uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Amine Kharrat

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.Sc. (Systems Science)

GRADE / DEGREE

System Science

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

A Floating-point Analog-to-Digital Architecture for a Wide Range-Dynamic Acquisition System

TITRE DE LA THÈSE / TITLE OF THESIS

Voicu Groza

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Emil Petriu

Tet Yeap

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

A Floating-Point Analog-to-Digital Architecture for a Wide Range-Dynamic Acquisition System

By
Amine Kharrat

A Thesis
Presented to School of Graduate Studies and Research
In partial fulfillment of the
Requirements for the degree of

Master of Science

Master program in Systems Science
University of Ottawa

Ottawa, Ontario, Canada, 2009

Your file Votre référence
ISBN: 978-0-494-61147-0
Our file Notre référence
ISBN: 978-0-494-61147-0

Abstract

In this thesis, the floating-point analog-to-digital conversion approach has been investigated for the implementation of a computer acquisition system for wide range-dynamic signals.

The Floating-point A/D converters (FP-ADC) can dissociate the resolution from the wide range and thus an ADC with a moderate resolution can achieve a wide dynamic range.

This dissertation studies the floating-point analog-to-digital converters and describes their characteristics and specifications. More specifically it analyses the sequential FP-ADC and improves the state of the art in conversion time as well as precision in floating-point analog-to-digital converters. It shows an implementation of the sequential FP-ADC on a daughter card which is connected to a Stratix Field Programmable Gate Array (FPGA) from Altera. The correctness of the design was verified by computer simulations while the functionality of the implemented FP-ADC on the manufactured 4-layer Printed Circuit Board (PCB) was tested on a test bench controlled by a PC.

Acknowledgments

This thesis would have been impossible to complete without the support and help I received. I would like to acknowledge those who have provided me with the opportunity to finish this thesis and my graduate studies.

First I would like to thank my supervisor and professor at the School of Information Technology and Engineering at the University of Ottawa, Dr. Voicu Groza. I am deeply grateful for his guidance and support and his devotion and patience with me during the course of my research.

I would also like to thank the members and staffs at the University of Ottawa and all parties who have been involved in this process especially Analog Devices for providing me with samples.

And finally I would like to express my gratitude and appreciation to my family and friends for their understanding and encouragements for the period of my graduate studies.

Acronyms

ADC	Analog-to-Digital Converter
CMOS	Complementary Metal-Oxide Semiconductor
DAC	Digital-to-Analog Converter
DNL	Differential Nonlinearity
DR	Dynamic Range
DSP	Digital Signal Processing
ENOB	Effective Number of Bits
FP-ADC	Floating Point Analog-to-Digital Converter
FPGA	Field Programmable Gate Array
INL	Integral Nonlinearity Error
LSB	Least Significant Bit
MSB	Most Significant Bit
PCB	Printed Circuit Board
PDF	Probability Density Function
PGA	Programmable Gain Amplifier
RMS	Root Mean Square
SNR	Signal Noise to Ratio
UART	Universal Asynchronous Receiver and Transmitter
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuits

Table of Contents

List of figures

List of tables

Chapter 1

1. Introduction

High performance devices such as oscilloscopes, hearing aids and digital radios need high precision components due to their advanced data processing and extensive analog to digital operations. Analog-to-Digital converters (ADC) are the key components which process the analog signal and convert it to a digital signal. Because of their importance in providing the link between the analog and the digital world, they are usually considered as the bottleneck in the system architecture of mixed-signal devices [1].

So as more applications move toward embedded digital signal processing capabilities and wide range dynamic signal, high resolution converters tend to be the logical solution. At the same time, considering that they are also increasingly complex to implement, it is simply not viable to keep increasing the resolution indefinitely and therefore this solution seems impractical and costly.

The important factors associated with ADCs are mainly related to the conversion time and the degree of precision they provide. Recently, the increased demand of low consumption components which lowered even further the actual input voltage and the continuous miniaturization of silicon based integrated circuits pushed to its limits the current technology. This led toward the exploration of newer, more precise and more flexible methodology to achieve the highest resolution and the fastest conversion possible with a broad range of signal inputs while also maintaining a minimal cost.

But, it's often not required to achieve a high degree of resolution at large amplitude. In other words, there is no need to provide excellent accuracy at all levels which is somehow difficult and complex. Instead, it's more realistic to provide more or less the same accuracy for small and large signal within the input range [2], [3]. This will help lower the production cost that comes with larger number of bits and over come some limiting factors like, noise, component mismatch and amplifier offset [1], [4].

The floating point analog to digital conversion approach dissociate the resolution from the wide range and thus an ADC with a moderate resolution can achieve a wide dynamic range.

1.1. Motivation

The motivation of this dissertation is to further improve the conversion time as well as the precision of the floating point analog-to-digital converter (FP-ADC) especially with low voltage input signals where the converters performance is most likely to be degraded due to noise amplification. The thesis provides a new architecture that investigates the relationships between accuracy, conversion time and power consumption by implementing a FP-ADC and providing a link with a Field Programmable Gate Array (FPGA). This method combines two ADC's that work sequentially: one acquires the exponent, while the second one, with a variable gain, finds out the best representation of the mantissa [5].The main obvious advantage of such technique is to lower the acquisition time while preserving the precision of the analog to digital conversion. Besides, the virtual input range of this ADC is proven to be much larger than the actual one so it can handle large as well as small signal input due to low power supply.

During the course of the research, this dissertation explores through the literature diverse technologies in analog-to-digital converters and identifies the advantages and disadvantages of different architectures available with the emphasis on Floating-Point ADCs. The objective of this work is to implement and provide a circuit for the FP-ADC that improves the conversion time and maintains a good enough resolution with a higher level of signal to noise ratio (SNR) for the acquired quantized signals especially on its low range. As a consequence, we implemented a new FP-ADC on a typical hardware interface. We used for that matter a daughter card connected to an ALTERA board to place and route the FP-ADC. Prior to that, a comprehensive simulation was undergone to ensure the correctness of the design and later on the manufacturing of the printed circuit board which is the FP-ADC. Testing was carried out in the next phase to provide an estimate of the performance of the developed design and validate the work.

1.2. Contribution

The dissertation presents a FP-ADC converter which provides high relative precision when quantizing high dynamic signals. A prototype was conceived with a 16-bit FP-ADC that is designed to operate at a rate of 125 KHz which makes it suitable for instrumentation and measurements specifically in the biomedical engineering field. This dissertation starts with a review of the state of the art techniques, a study of the concept and an investigation of a possible implementation of the FP-ADC. Then, it proposes a design of a reference FP-ADC architecture utilizing the analysis of the proposed FP-ADC as the basis. The new model is a foundation which in turn provides a test bed for analyzing and studying the characteristics of the FP-ADC. This research intends to offer a fully flexible solution using off the shelf components and a series of a conceived testing

methods which are believed to make it suitable for a wide variety of applications and to serve as the starting point for research of improved future designs of the FP-ADC.

In summary, this dissertation shows that improvements to the sequential floating-point ADC architecture can be made as it provides an increase in resolution, overall dynamic range and conversion time. A number of possible ameliorations and drawbacks are discovered while remedies are suggested.

1.3. Thesis overview

The outline of the thesis is as follow:

Chapter 2 introduces the fundamentals. It reviews the IEEE 754 standard for floating point arithmetic and the IEEE 1241 standard for terminology and test methods for analog-to-digital converters. It also gives an insight on the ADCs specifications and characteristics. And finally, it introduces the floating-point quantization process and discusses the effect of the added noise on it

Chapter 3 includes a summary of the theory knowledge used through this work. It presents the different ADC architectures and a critique of the state-of-the-art in floating point analog-to-digital conversion. It goes through a review of the floating-point A/D converters before presenting the proposed approach used in this thesis.

Chapter 4 presents the architecture of the system being investigated by this thesis that is the sequential architecture of the floating-point A/D converter. It also covers the different interactions between the ALTERA board and the daughter card on which the FP-ADC is implemented.

Chapter 5 presents the simulations and testing of the FPGA and the FP-ADC subsystems. It also discusses the design choices based on the results.

Finally, Chapter 6 concludes this work and gives a summary on the important results we established before the introduction of future works and enhancements.

Chapter 2

2. Fundamentals

This chapter begins with basic analog-to-digital converter terminology and a comparison between the floating-point and the fixed-point representation followed by an introduction of the different standards used through the thesis. Then it will elaborate more on the ADC specifications and characteristics. And finally, it will introduce the quantization process and more specifically the floating-point quantization and the effect of the added noise on it.

2.1. Floating-point representation versus fixed-point representation

In digital signal processing (DSP) two representations exist: the fixed point representation and the floating point representation. These representations describe how the numbers are being represented within electronic devices. A typical 16 bit fixed point DSP is capable of representing 2^{16} different bit patterns whereas a typical floating point DSP of 32 bit length is capable of representing 2^{32} bit patterns which shrunken the gap between two adjacent numbers. Additionally, in floating point notation the represented numbers are not uniformly spaced so the gap between two numbers is proportional to the numbers themselves and therefore is sensibly larger between large numbers and smaller between two small numbers. This gives an advantage in precision to the floating point representation over the fixed point representation due to better Signal-to-Noise Ratio (SNR); a stored number is rounded up or down to the next value consequently a smaller gap implies a little added noise or less quantization noise. Another advantage of the

floating point representation is the greater dynamic range due to the exponentiation and the superior exactitude in the internal representation of data: for example a 16 bits by 16 bits multiplication in fixed point DSP requires 16 bits + 16 bits + 8 bits overflow for the total intermediate product where as the floating point DSP keeps only the most significant 32 bits and still manage a better accuracy. These benefits don't come without a cost though. The more complex internal circuitry of the floating point DSPs, the wider data path 32 bits versus 16 bits and the greater number of Inputs/Outputs dictated by the wider data bus requires a bigger package and a larger die than the logic will use. This translates inexorably into a significant cost premium. However, because floating point DSPs were easier to program they were adopted early on for low volume applications where time and software development cost were critical. Today, the prohibitive cost of the floating point DSP tends to diminish as the technology permits to fit more transistors on the same space as before and more and more applications move toward floating point. In the long term, the Floating point DSPs will benefit from large scale manufacturing and therefore is predicted to become more popular within the developer community [6], [7].

The following graph depicts the process of digital signals converted from analog signal by the ADC and outputted back in analog form by the DAC.

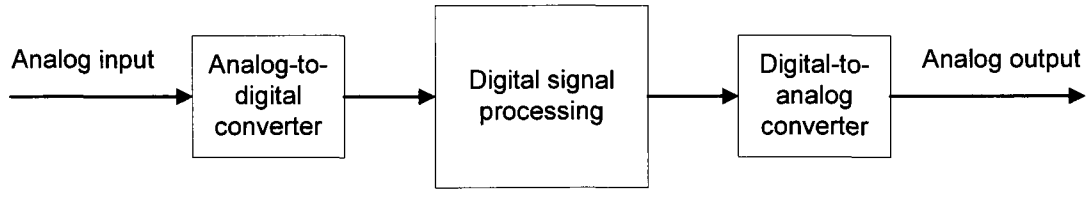

Figure 2-1 A simple digital processing system [8]

In a typical electronic control system, the analog signal originates from a transducer or other type of sensors. They measure and convert a physical phenomenon such as temperature to electric signal. The electric signal can be an analog current or voltage which is proportional to the phenomenon being measured. The analog signal is then converted to digital by the ADC and sent to the DSP unit where some processing and optimization algorithms are used. Once converted back by the DAC to analog, the signal is ready to trigger an analog actuating device such as valves or switches to maintain a certain temperature on a heating system for example.

2.2. Floating-point arithmetic standard

The IEEE 754 standard [13], fruit of the collaboration of the Floating-Point working group within the IEEE standard committee, defines a family of commercially feasible ways for new systems to perform binary floating point arithmetic and is probably the most widely used standard when it comes to floating point computation in Floating Point Unit (FPU) or Central Processing Unit (CPU).

The above standard was created in 1985. Most recently, in 2006 and 2007, an attempt was made to rewrite that standard but those drafts left unapproved. Therefore, we will focus on the 1985 version of it.

The standard encompasses four formats for representing floating point values separated in two groups: basic and extended and with two different widths: single and extended. So the four formats are as follow: single-precision (32 bits), double-precision (64 bits), single-extended precision (more than 43 bits) and double-extended precision (more than 79 bits). The 32 bits single precision format is the required format though.

The standard also specifies four rounding methods: Round to the nearest, the default rounding method, and three directed rounding: round toward positive infinity, round toward negative infinity and round toward Zero. Operations like addition, subtraction, multiplication and division are part of the standard as well.

There is also a definition for special values: infinity, NaNs and signed Zero and five exceptions types (including how to detect them and how to deal with them). Among these exceptions, the following: invalid operation, division by Zero, overflow, underflow and inexact result.

2.2.1. General layout

The IEEE standard floating point numbers are divided into three basic components: The sign bit, the mantissa (composed of the fraction and an implicit leading bit) and the biased exponent.

The figure below explains the repartition of these components in a single precision (32-bit) format. Here bits 0 through 22 represent the fraction, bits 23 to 30 correspond to the exponent and the last bit 31 is the sign bit.

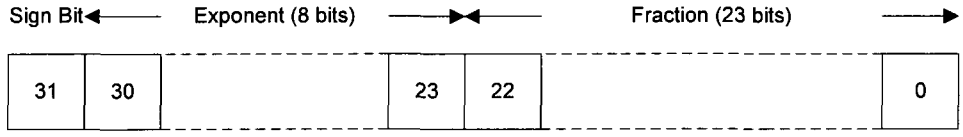

Figure 2-2 IEEE 754 Format of Floating Point Numbers

The number representation is then:

Number = Sign * 2E * Mantissa

Where Sign = 0 for positive numbers and 1 for negative numbers

E = Exponent -127 (the bias)

Mantissa = 1.Fraction (as the first bit is assumed to be 1)

2.3. Analog-to-digital converter background

The IEEE standard 1241-2000 [9] defines the terms, definitions and test methods used to specify, characterize, and test analog-to-digital converters. The standard considers only ADCs with quantized and sampled output values (discrete values at discrete times).

Many methods exists for representing a continuous analog signal as a discrete sequence of binary words, nevertheless, it is usually assumed that the relationship between the input signal and the output values approximates the staircase transfer curve depicted on the following part a) of the figure.

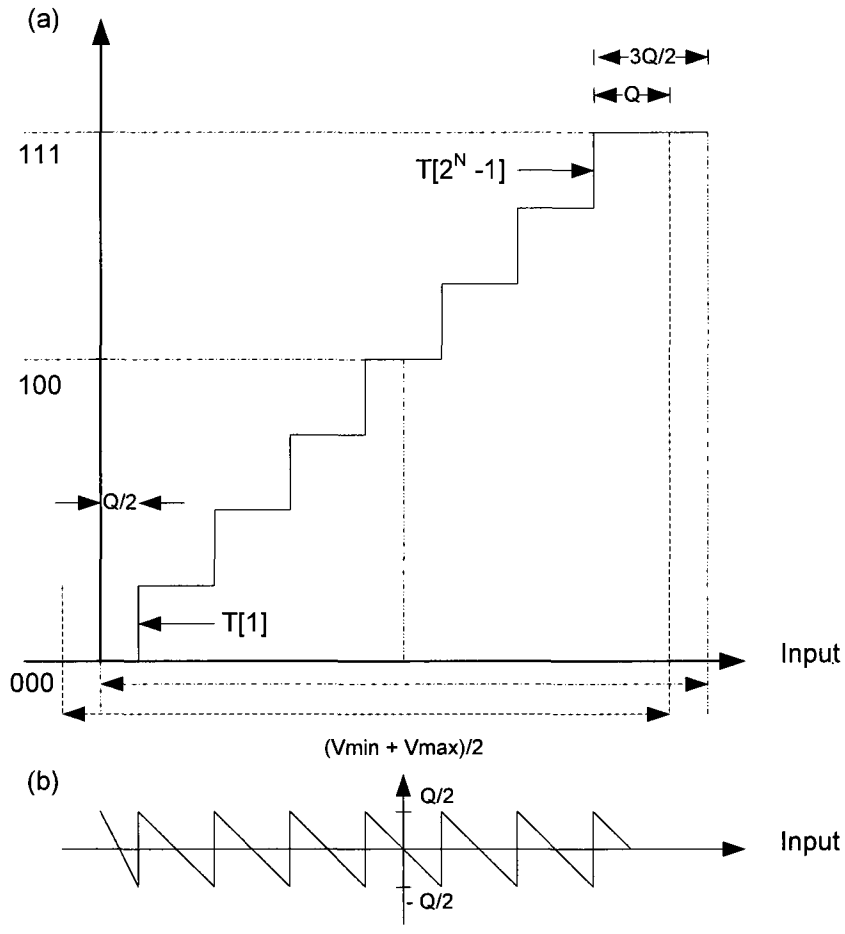

Figure 2-3 Staircase ADC Transfer Function

In this model, the full scale input range (FS) at the ADC from V_{min} to V_{max} is divided into uniform intervals, known as code bins, with nominal width Q. The number of code transition levels in the discrete transfer function is equal to $2^N - 1$, where N is the number of digitized bits of the ADC. Also in this model, two conventions exist for relating V_{min} and V_{max} to the nominal transition points between code levels, mid-tread and mid-riser.

The dotted lines in the figure at V_{min}, V_{max} and $(V_{min} + V_{max})/2$ indicate the mid-tread convention. In this case, the first transition occurs at Q/2 after V_{min} and the last transition occurs at 3Q/2 before V_{max}. The midpoint of the range, $(V_{min} + V_{max})/2$ occurs

in the middle of the code, i.e., on the tread of the staircase transfer function, hence the convention's name mid-tread.

The dashed lines, however, at V_{min}, V_{max} and $(V_{min} + V_{max})/2$ indicate the mid-riser convention. In this case, the first transition occurs at Q after V_{min} and the last transition occurs at Q before V_{max}. The midpoint of the range, $(V_{min} + V_{max})/2$ occurs on a staircase riser. The difference between the two conventions is a displacement along the voltage axis by an amount of Q/2. This displacement has no effect on the result and either convention may be used. It's important to note though that even in an ideal ADC, the quantization process produces errors. These errors contribute to the difference between the actual transfer curve and the ideal straight-line transfer curve, which is plotted as a function of the input signal in part b) of the figure.

In addition to the nominal quantization error, ADCs have different other errors which are categorized into static and dynamic errors depending on the rate of change of the input signal at the time of digitization. A signal is considered static if it is slowly varying and its effects are equivalent to those of a constant signal. Static errors, including the quantization error mentioned earlier results from the non-ideal spacing of the code transition level. Dynamic errors occur because of additional sources of error induced by the time variation of the analog signal being sampled. Sources include harmonic distortion from the analog input stages, signal-dependent variations in the time of samples, dynamic effects in internal amplifier and comparator stages, and frequency dependent variation in the spacing of the quantization levels.

2.4. General specifications of an analog-to-digital converter

The design of an analog-to-digital converter that converts continuous time signals into discrete time starts with an analysis of the overall system requirements and involves a choice between various trade-offs. Consequently, it is important to define and prioritize the key characteristics so that the system meets the specifications and performs its intended function. The first requirements that the system must comply with are the dynamic range and the speed of conversion. Then come the required accuracy and operating power dissipation along with other functional and physical requisites. Before defining the characteristics of A/D converter, the next section gives a brief overview of the relationship between A/D and D/A conversions.

2.4.1. Analog/Digital conversion relationship

The translation of an analog quantity is shown bellow on the graph of the transfer function of an ideal 3-bit ADC. The resolution of an ADC, determined by the number of bits in the digital output, is the smallest quantizing step size. That is the range associated with analog input voltages over which the ADC will produce a given output code is called a quantization uncertainty and is equal to 1 LSB. In theory, it is assumed that the width of the transition region between adjacent codes is zero. But, in practice the width is non-zero due to transition noise involving these levels. From Figure 2-4, the corresponding output code to an analog input is defined by the code center which lies in the middle between two adjacent transitions. For example, the A/D decision points for choosing 101 digital output goes from 101 - ½ LSB to 101 + ½ LSB. Assuming perfect operation, no error in the circuitry of the A/D converter, the conversion dynamic range and accuracy is limited

to ± ½ of its LSB. The full scale analog input voltage is 7/8 FS or FS – 1LSB which is a common convention in data conversion notation and applies for both ADCs and DACs.

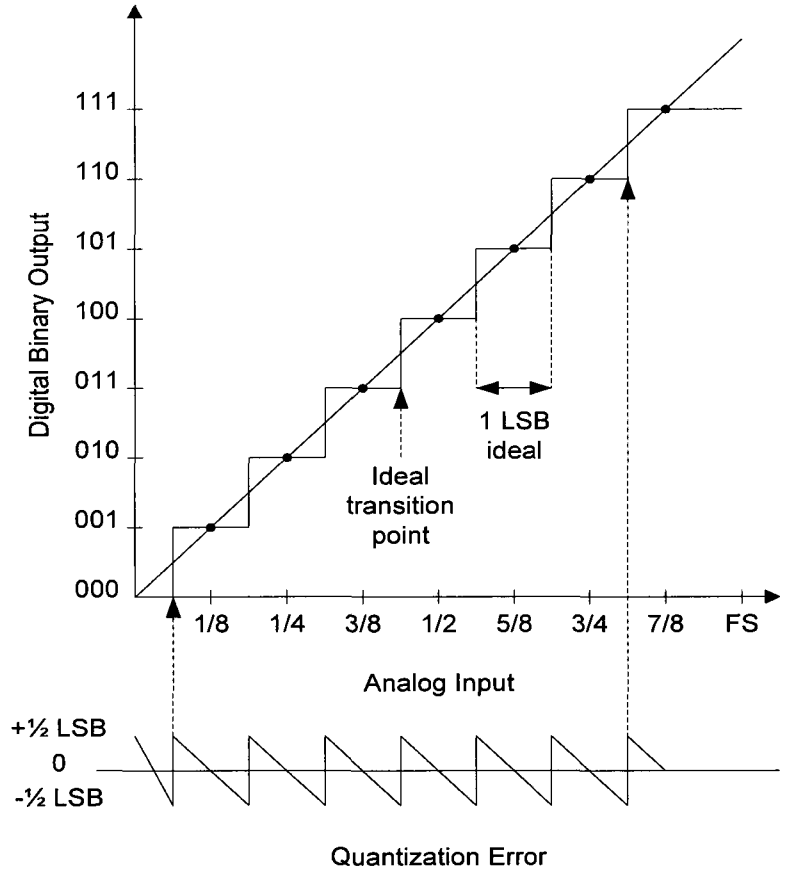

Figure 2-4 Ideal 3-bit A/D converter

The D/A converter analog output is also limited in resolution by the number of bits the D/A can convert to an analog value and therefore the step sizes of the discrete analog output values are equal to the LSB as shown in Figure 2-5.

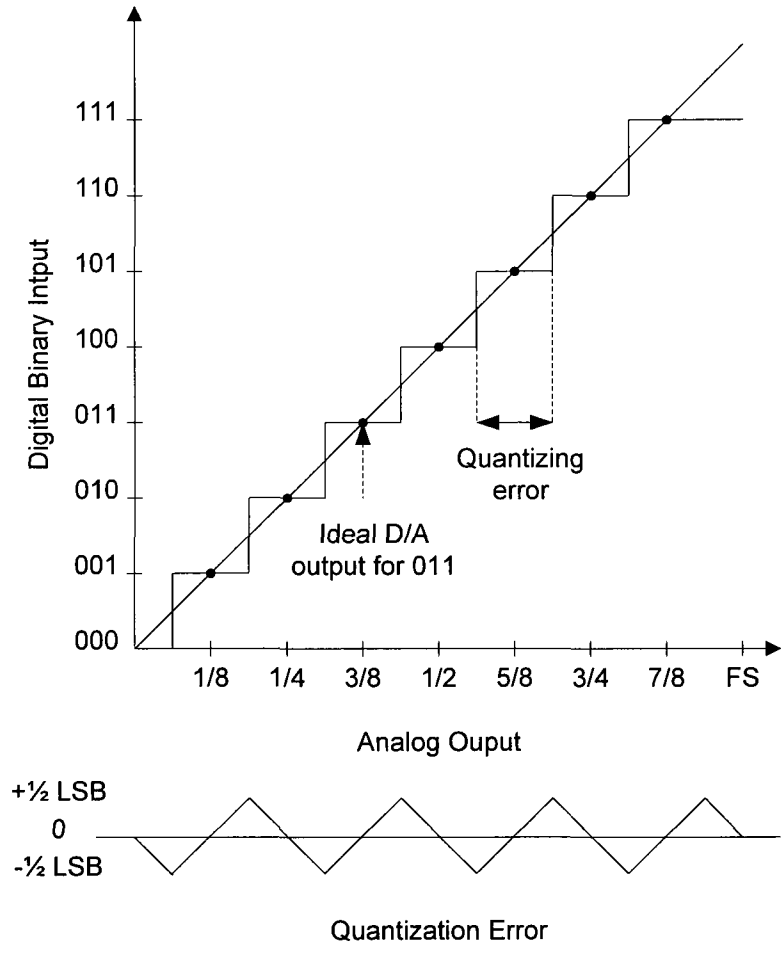

Figure 2-5 Ideal 3-bit D/A converter

In data conversion systems, the coding methodology must be related to the analog input range of the ADC or the analog output range of the DAC. The most popular code used, given that the input is a positive signal (i.e., positive voltage), is straight binary as shown in Table 2-1 for a 3 bit converter. There are 8 distinct possible levels ranging from 000 to 111. Note that the highest level representing 111 is not FS but FS – 1 LSB. The table below shows the base-10 equivalent number, the value of the base-2 binary code relative to full scale (FS) and also the corresponding voltage for each code.

Table 2-1 Binary Codes, 3-bit Converter

BASE 10 NUMBER	SCALE	+5V FS	BINARY
+7	+FS – 1 LSB = +7/8 FS	4.375	111
+6	+3/4	3.750	110
+5	+5/8	3.125	101
+4	+1/2	2.500	100
+3	+3/8	1.875	011
+2	+1/4	1.250	010
+1	+1/8	0.625	001
0	0	0.000	000

2.4.2. Characteristics of an analog-to-digital converter

In this section, the operational characteristics of the analog-to-digital converter and the accuracy parameters are discussed. Nevertheless, as a system designer, the best approach to design a conversion system, regardless of the detail of given requirements, is to understand the overall operation of the whole circuit.

2.4.2.1. Accuracy parameters

As discussed earlier, there is a tradeoff between cost, accuracy, power consumption and so forth. The design of an ADC involves some choices to be made since they are the primary elements in determining the accuracy of a measurement system. Quite often, relative accuracy is more relevant than absolute accuracy which is the magnitude of the deviation from the actual to the ideal transfer function of an ADC.

Absolute accuracy represents the accumulation of various errors; for example an ADC with gain error of 0.5 LSB and differential non-linearity of 0.5 LSB may have up to 1 LSB absolute accuracy error. To alleviate the problem of having larger errors for larger values and smaller errors for smaller values which basically requires many bits in the number that describes those values, relative accuracy comes in handy. As in many cases it is not required to have full accuracy at all magnitudes but rather keep the accuracy more or less constant then FP-ADCs are the solution.

Before being converted analog signals are generally amplified. This is necessary to raise the signal to a level i.e., the upper half level of the input range set by high and low voltage reference, so that it can be effectively converted by the A/D converter. However, this also amplifies the noise which is inherent to the amplifier circuitry and affects various characteristics within the ADC.

Dynamic range: The value of the noise at small signal levels or the uncertainty of analog measurement compared to the full-scale signal value at the upper limit of its range determines the dynamic range of the usable analog signal. More generally, the dynamic range is the ratio of the largest input that can be converted to the smallest step size of the converter. For example a 3-bit ADC with an input range of zero to five volts has a quantization step size of $(5V - 0V) / 2^3 = 0.625V$. So the dynamic range is 5 V / 0.625 V = 8 which can also be expressed in decibels as 20 log 8 = 18 dB or in power 2 format, that is a 3-bit ADC.[11]

Quantization step: The ADC converts the signal into discrete output levels using a nonlinear process. As it shows from the previous example, the smallest discrete step is called the quantization step, which is a function of the ADC resolution, or number of bits.

Resolution: The accuracy in FP-ADC is called resolution. In our example, the 3-bit ADC has 8 levels in which the input signal can lies within. This ADC is said to have an 8-bit resolution. In fact the definition states that the resolution refers to the number of quantization levels an input signal can be determined to and that this later number is usually given in bits.[11]

Quantization error: Quantization error is the natural error that occurs during the conversion of a signal from analog to digital. The error is due to the difference between the magnitude of the analog input and the digital code because the same output code can designate a range of analog input. In Figure 2-4, the sawtooth plot at the bottom shows the ideal difference in magnitude between analog input and the resulting digital code for the entire input range of the converter. This difference is the quantization error. It is clear that any analog voltage that falls between two transition points will have an output code that is inaccurate by up to ½ LSB at the worse case.[11]

Signal-to-noise Ratio: Simply stated the signal-to-noise ratio, for a waveform perfectly reconstructed from digital samples, is the ratio between an RMS (Root Mean Square) full-scale analog input and its RMS quantization error. The RMS value of a sine wave is one half its peak-to-peak value divided by $\sqrt{2}$, and the quantization error is the difference between an analog waveform and its digital counterpart, which is the staircase-shaped transfer curve from Figure 2-4. So the RMS value of the quantization error is its peak value (1/2 LSB) divided by $\sqrt{3}$. Therefore for an ideal N-bit converter SNR is defined as:

$$SNR = 2^N \times \frac{\sqrt{3}}{\sqrt{2}} = 1.23 \times 2^N \qquad (2\text{-}1)$$

In terms of the logarithmic decibel scale, useful for signals with wide dynamic range, SNR is expressed as:

$$SNR_{dB} = 6.02 \text{ x } N + 1.76 \tag{2-2}$$

Effective Number of Bits: Effective number of bits (ENOB) is a measure of overall A/D performance under dynamic conditions. Because of quantization noise, an ideal N-bit ADC will still have an effective number of bits that is less than N.

Moreover, the sum of error sources like quantization error, dynamic differential nonlinearity error, missing codes, integral non linearity, jitter, and noise contributes to a lower effective number of bits. ENOB can be expressed in term of SNR as:

$$ENOB = \frac{SNR - 1.76}{6.02} \tag{2-3}$$

For an ideal 3-bit ADC, $ENOB = \frac{18dB - 1.76}{6.02} = 2.7bits$

Differential Nonlinearity Error: Abbreviated as DNL error is a measure of how uniform the transfer function step sizes are. The difference in magnitude between each step size and the ideal step size is DLN error.

$$DNL = 1 - \frac{V(x) - V(x+1)}{LSB} \tag{2-4}$$

Where code bin width = V(x) – V(x+1) and LSB is the ideal spacing for two adjacent output codes. Note that a DNL error specification of less than or equal to 1LSB guarantees a monotonic transfer function with no missing codes.

Integral Nonlinearity Error: INL measures the deviation of the line joining the code midpoints from a straight line or the ideal location.

$$INL = \frac{Vx - Vzero}{LSB} - X \tag{2-5}$$

Where Vx is the analog value represented by the digital output code X, Vzero is the minimum analog input corresponding to an all-zero output code and LSB is the ideal spacing for two adjacent output codes.

Gain Error: Gain error, also known as gain flatness, is the difference between the dynamic gain, G (f), of the ADC at a given frequency and its gain at a specified reference frequency, divided by its gain at the reference frequency. It is the error contribution due to the displacement of the actual ADC characteristic from the reference one.

$$E_{G(f)} = \frac{G(f) - G(f_{ref})}{G(f_{ref})} \times 100\% \qquad (2\text{-}6)$$

Where f_{ref} is the chosen reference frequency.

Offset Error: Offset error identifies the deviation from the ideal location of the lowest transition level on the ADC transfer function.

The Figure 2-6 bellow summarizes the effect of these errors: DNL, INL, gain and offset error.

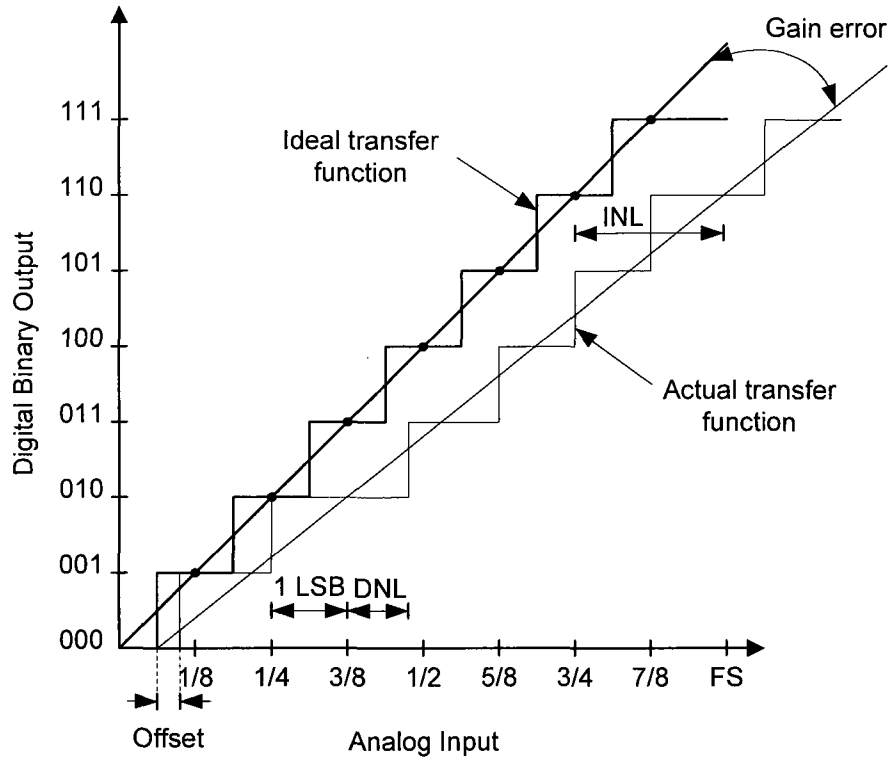

Figure 2-6 Illustration of DNL, INL, gain error and offset

2.4.2.2. Operational Characteristics

Some of the ADC operational characteristics used in this thesis and already mentioned in preceding sections including

Full scale range: The full scale input range (FS) at the ADC is the input range of voltages from V_{min} to V_{max} over which the ADC will digitize the input.

Input bandwidth: Relates to the maximum frequency analog signal an ADC can convert and still meet its performance specifications. An ADC is usually required to convert input signals with bandwidth of up to a half their sampling rate or Nyquist rate.

Least significant bit: LSB is the magnitude of the ADCs transfer function's step size. It refers to the bit of the digital output code that has the smallest weight. LSB is found using the following:

$$\text{LSB} = \text{full-scale range} / (2^N - 1) \tag{2-7}$$

2.5. Quantization

The process of quantization is the mapping of a continuous signal into a discrete value. The analog signal V_{in} is expressed in term of quantized value d and the quantization error ε as follow:

$$V_{in} = d + \varepsilon \tag{2-8}$$

The quantized value chosen depends on the rounding. For an A/D converter the rounding is chosen in such a way that it minimizes the quantization error which means preferably rounding to the nearest quantized value.

The floating-point representation is suitable for covering a very wide dynamic range with relatively small number of digits. Nevertheless, due to the quantization of the input to a finite accuracy, there are round off errors which are inevitable and roughly proportional to the amplitude of the represented quantity. To put this in perspective, in the case of uniform quantizers, round off errors are bounded between ±q/2 and thus are not in any way proportional to the represented quantity.

2.5.1. Fixed-point quantization

In fixed-point number representation, an array of bits is used, the binary word w, to represent a number. For a word of length n bits, the number is expressed as

$$w = \sum_{k=0}^{n-1} b_k 2^k \tag{2-9}$$

Where b_k is the bit at position k in the binary word. The total range of values is 2^n. The resolution is the smallest step between unique values also referred as LSB.

The quantized value d can be represented by a binary word using the following relationship

$$d = w.\frac{V_{FS}}{2^n} \tag{2-10}$$

Where VFS is the full scale input range. The smallest step d is equal to

$$q = \frac{V_{FS}}{2^n} \tag{2-11}$$

Where q is equivalent to one LSB. The quantization error varies between –q/2 and +q/2. It is known that the quantization error has a white spectral density with a rectangular probability density function (PDF) as shown in Figure 2-7 under the assumption that the input is a random signal with the standard deviation $\sigma_S >> q$.

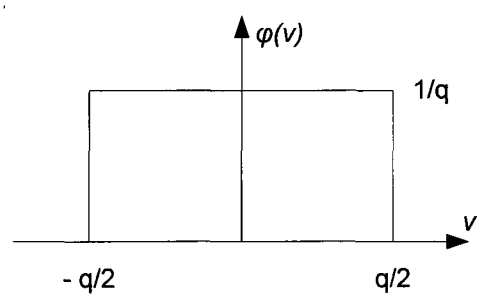

Figure 2-7 PDF of quantization error

The energy of the quantization error also called quantization noise is compared to the signal power to obtain the expression of the SNR. The power of the quantization error translated to the variance of the error is

$$\bar{\varepsilon}^2 = \frac{1}{q}\int_{-\frac{q}{2}}^{\frac{q}{2}} v^2 dv = \frac{1}{12}q^2 = \frac{V_{FS}^2}{12.2^{2n}} \tag{2-12}$$

The SNR is obtained by comparing the power of the error to the power of a full scale sinusoid ($\frac{V_{FS}^2}{8}$)

$$SNR = \frac{3}{2}2^{2n} = 6.02n + 1.8db \tag{2-13}$$

For non sinusoidal inputs, the signal is modeled as a stochastic process with some PDF. The Laplace distribution is used for a single signal input and the Normal distribution is used for compound signals. The SNR for fixed-point quantization is then

$$SNR = 6.02n + 10.8 - 20\log_{10}(\frac{V_{FS}}{\sigma(v_{in})})db \tag{2-14}$$

Where $\sigma(v_{in})$ is the standard deviation of the input signal distribution.

2.5.2. Floating-point quantization

Floating-point numbers are described by the mantissa M and the exponent E therefore quantized numbers can be expressed as

$$d = M2^E.\frac{V_{FS}}{2^{n+2^{n_e}-1}} + \frac{V_{FS}}{2} \tag{2-15}$$

Where n and n_e are the word lengths of the mantissa and the exponent respectively. $V_{FS}/2$ is the input voltage corresponding to a zero digital signal.

The mantissa may represent both negative and positive values so the most significant bit is a sign bit.

$$M = \sum_{k=0}^{n-1} b_k 2^k - 2^{n-1} + 0.5 \tag{2-16}$$

Where the offset between negative and positive numbers is cancelled by adding 0.5.

The smallest step in the case of floating-point number deduced from the equation of d is

$$q = \frac{V_{FS}}{2^{n+2^{n_e}-1}} \tag{2-17}$$

It varies with the signal amplitude therefore the PDF is not uniform and the PDF of the relative quantization error is used.

A floating-point quantizer is illustrated in Figure 2-8. The input to this quantizer is x, a variable that is generally continuous in amplitude. The output is x', a variable that is discrete in amplitude over a floating-point number scale. The input-output relationship is a staircase function that does not have a uniform steps.

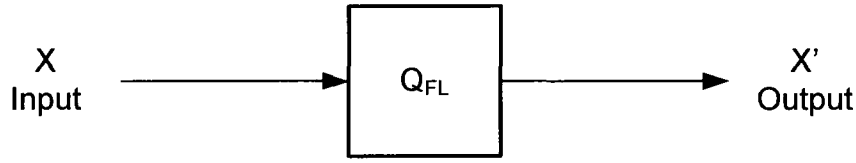

Figure 2-8 A Floating-point quantizer

The input-output relationship for a floating-point quantizer with a 3 bit mantissa is represented in Figure 2-9. The input variable is x and its floating-point representation is x'. The smallest step size is q. The spacing of the cycles (when the exponent is increased by 1) is determined by the parameter Δ and given by:

$$\Delta = 2^P q \tag{2-18}$$

Where p is the number of bits of the mantissa. With a 3 bit mantissa, $\Delta = 8q$.

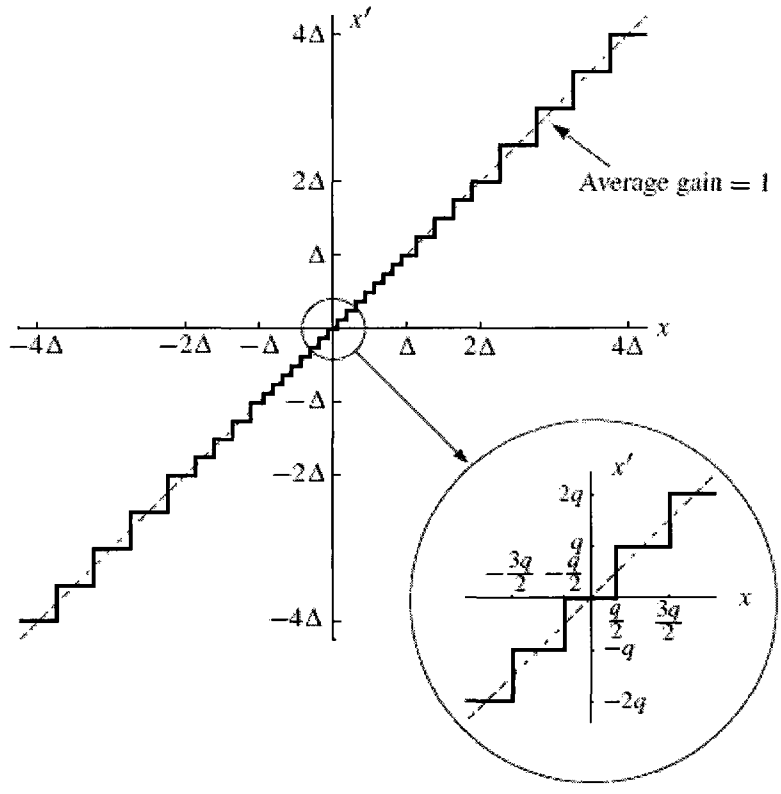

Figure 2-9 Input-output staircase function for a floating-point quantizer with a 3-bit mantissa [15]

The SNR is almost constant for a sinusoid signal. With n bits maximum for the mantissa the SNR of floating-point is equal to fixed-point quantization

$$SNR = 6.02n + 1.8db \tag{2-19}$$

For non sinusoidal input, the power of the relative quantization error for most distributions of the input signal is

$$\bar{\varepsilon}^2 = \frac{2^{-2(n-1)}}{8\ln 2} \tag{2-20}$$

The relative error has a white spectral density function and is uncorrelated to the input signal for most cases so the SNR for a floating-point quantization for practically most distributions of the signal is

$$SNR = 6.02n + 1.4db \tag{2-21}$$

The dynamic range DR is different from the SNR for floating-point quantization because the smallest step is much smaller compared to fixed-point quantization. Therefore, the dynamic range is

$$DR = \frac{3}{2}2^{2(n+2^{n_e}-1)} = 6.02(n+2^{n_e}-1)+1.8db \tag{2-22}$$

2.5.3. Floating-point quantization noise

The block diagram of the floating-point quantizer in Figure 2-10 shows the effect of the added noise on the quantization process. The actual round off noise is the difference between the quantizer output and the input and can be written as:

$$V_{FL} = x' - x \tag{2-23}$$

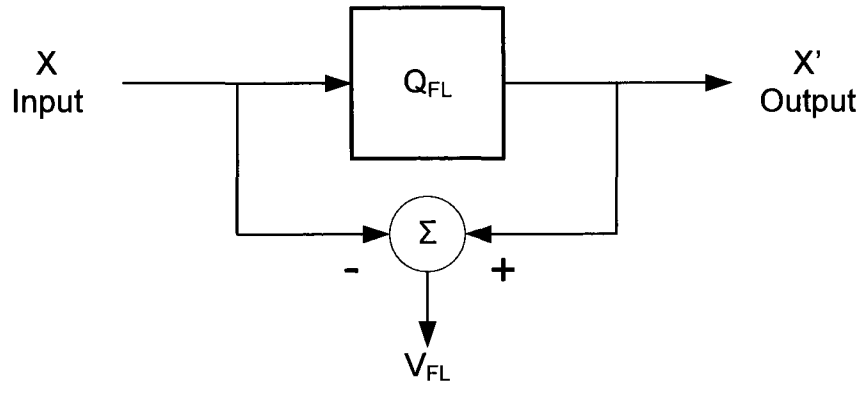

Figure 2-10 Floating-point quantization noise

The probability density function of the quantization noise represented by the pyramidal shape is not uniform because the staircase steps are not uniform.

The PDF of floating-point quantization noise with a zero-mean Gaussian input with a standard deviation $\sigma_x = 32\Delta$, and a 2 bit mantissa is shown in Figure 2-11:

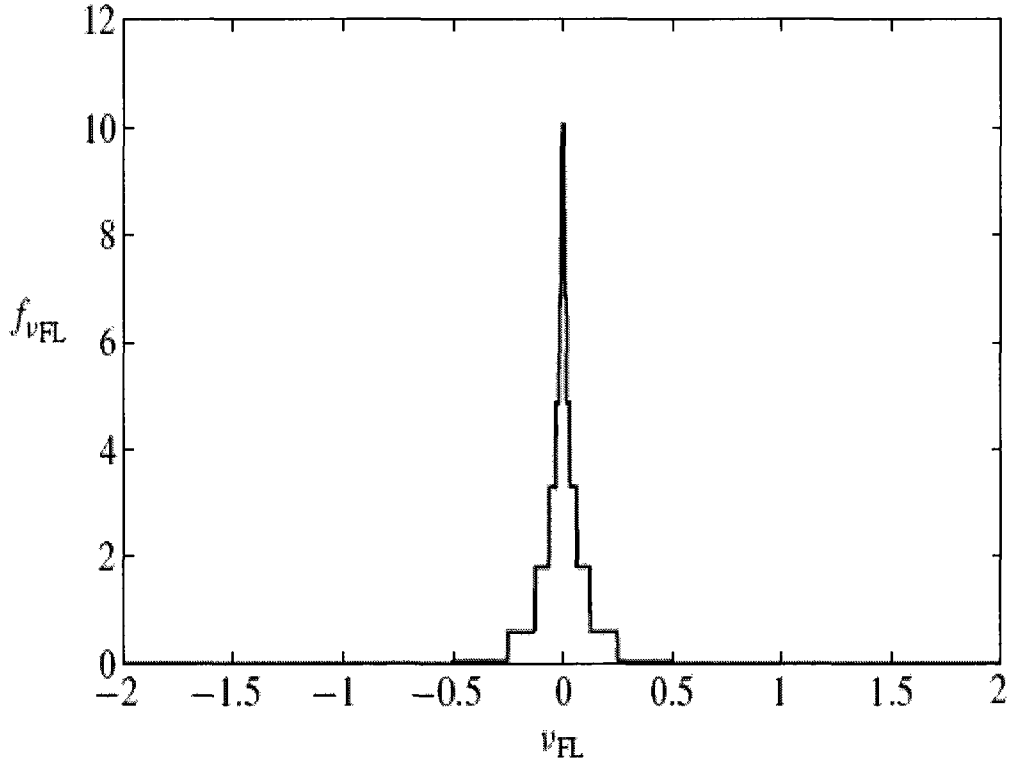

Figure 2-11 The PDF of floating-point quantization noise [15]

Chapter 3

3. Quantization of a wide range dynamic signals

This research work provides a new architecture of implementing a FP-ADC. It combines two ADC's that work sequentially: one acquires the exponent, while the second one, with a variable gain, finds out the best representation of the mantissa. It employs uniform quantizers to keep the precision of the sequential floating-point analog-to-digital converter and minimizes the conversion time close to the characteristics of the non-uniform one cycle quantizer.

This chapter introduces the state-of-the-art in floating point analog-to-digital conversion. It also presents the theory knowledge of this type of floating-point A/D converters and describes their characteristics and specifications. The chapter concludes by revealing the proposed FP-ADC architecture.

3.1. ADC architecture

Today, ADC applications can be classified according to five broad market segments:

1) Data acquisition,

2) Precision industrial measurement,

3) Voice band and audio,

4) High speed (with sampling speed greater than 5 MS/s) and,

5) Control loop applications where the ADCs are part of a feedback loop.

Most of these applications can be implemented using non-linear ADCs such as: successive-approximation (SAR) converters, Flash converters, and pipelined ADCs to name a few.

3.1.1. Integrating ADC

Also called ramp-compare ADCs provides high resolution and can rejects both line frequency and noise. The basic idea is to connect the output of a free-running binary counter to the input of a DAC, then compare the analog output of the DAC with the analog input signal to be digitized and use the comparator's output to tell the counter when to stop counting and reset. The effect of this circuit is to produce a DAC output that ramps up to whatever level the analog input signal is at, output the binary number corresponding to that level, and start over again. These ADCs have two main disadvantages: they suffer from a variation in the update frequency (sample time) which is longer for high input voltages and close-spaced for low signal levels plus they have a slow sample rate of analog signals because they use a counter that counts all the way from 0 at the beginning of each count cycle which is not the best counting strategy.

3.1.2. Flash ADC

Also called parallel converters are simple circuits formed of a series of comparators, each one comparing the input signal to a unique reference voltage. The advantage of their simplicity in terms of operational theory makes them the fastest and most efficient ADCs as they are limited only in comparator and gate propagation delays. Therefore, these ADCs are capable of operating from hundreds of Mega samples per second (MS/s) to tens of Giga samples per second. Flash ADCs are unfortunately the

most components intensive for any given number of output bits. An N-bit converter requires 2^{N-1} comparators and a total of 2^N resistors which provide the reference voltage.

Flash ADCs are mainly suitable for high bandwidth consuming applications. However these converters tend to consume a lot of power, have relatively low resolution and can be quite expensive. This limits them to very high frequency applications like video and wideband communications that typically cannot be addressed any other way.

3.1.3. Logarithmic ADC

A Logarithmic ADC is used because a Linear ADC cannot cope with the needed high resolution imposed by systems with a wide dynamic range. Since many systems accept that the resolution is proportional to the magnitude of the signal, it is possible to use a logarithmic like function to compress the signal. Therefore a logarithmic ADC with a non-uniform quantization can be implemented. The process of compressing and expanding is called companding and the function referring to compressing and expanding is called the codec.

The codec according to the c-law is:

$$f_c(x) = sign(x)\frac{\ln(c|x|)}{\ln c} \tag{3-1}$$

Where c is the compression coefficient and $1/c \le |x| \le 1$.

For signals close to zero, there exists two codecs. The codec according to the μ-law is defined as:

$$f_\mu(x) = sign(x)\frac{\ln(1+\mu|x|)}{\ln(1+\mu)} \tag{3-2}$$

Where μ is the compression coefficient.

The codec according to the A-law is composed of two parts; one linear part for small signals and one logarithmic part for large signals. It is defined as follow:

$$f_A(x) = \begin{cases} \dfrac{Ax}{1+\ln A}, |x| \le \dfrac{1}{A} \\ sign(x)\dfrac{1+\ln(A|x|)}{1+\ln A}, |x| > \dfrac{1}{A} \end{cases} \quad (3\text{-}3)$$

Where A is the compression coefficient.

The implementation of Logarithmic ADCs can be assigned to three categories:

1) Analog compression ADC, an analog logarithmic circuit followed by a Linear ADC.

2) Digital compression ADC, a high resolution converter followed by a digital compression

3) Logarithmic ADC, where the ADC and the logarithmic function are merged. The Logarithmic ADCs can be breakdown into three sub-categories which are: Sigma-Delta (Σ - Δ) converters, Pipelined ADCs and Successive Approximation ADCs.

3.1.4. Sigma-Delta (Σ - Δ) ADC

Also called over-sampling converters, they have a relatively simple structure. They consist of Σ – Δ modulator followed by digital decimation filter. The modulator, has an architecture which is similar to that of a dual slope ADC, it includes an integrator and a comparator with a feedback loop that contains a 1-bit DAC. The DAC is a simple switch that connects the comparator input to a reference voltage either negative or positive. A clock signal provides the modulator and the digital filter of the Σ – Δ ADC the proper timing [27]. Although slower than pipelined ADCs with a sampling rate in the

order of KS/s or hundreds of KS/s [12], the $\Sigma - \Delta$ ADCs offers four major advantages: low cost, low power, high resolution conversion and DSP compatible with the included digital filter in their conversion circuitry. This makes them popular in audio design and instrumentation with typical bandwidths less than 1 MHz and a range of 12 to 18 effective bits.

3.1.5. Pipelined and Sub ranging ADC

Despite Sub ranging ADCs being not as fast as their Flash counterpart with a speed greater than 100 MS/s [12], they use fewer comparators, draw less power, have lower input capacitance and can achieve higher resolution. For example a 4-bit Sub ranging ADC, that uses two 2-bit stages to digitize the analog input signal with the first stage converting the upper 2 bits and the second ADC converting the lower 2 bits, would require 6 comparators (3 for each ADC) comparatively to the 15 comparators required by the Flash ADC. Sub ranging ADCs are used on test equipments and oscilloscopes.

The pipelined ADCs originate from Sub ranging ADCs and are becoming the most popular ADCs. They have a similar speed around 100 MS/s with resolutions varying from 8 bits up to 16 bits [12] [29]. These ADCs have a digitally corrected Sub ranging architecture in which each of the two stages operates on the data for one half of the conversion cycle and then passes its residue output to the next stage in the pipeline prior to the next phase of the sampling clock. A track-and-hold between two stages serves as an analog delay line and allows more settling times for the internal Sub ranging ADCs, DACs and amplifiers. This gives the pipelined ADC a much higher sampling rate than a non pipelined one which proves to be ideal for applications such as ultrasonic medical imaging or digital receivers.

3.1.6. Successive Approximation Register (SAR) ADC

One method to overcome the integrated ADC's slow sample rate is to implement a special counter instead of the binary up counter used. The new counter is simply a successive approximation register hence the name SAR ADC. This register counts in a trial and fit fashion starting from the most significant bit and scanning all the values of bits until it reaches the least significant bit. This process gets a binary number that converges to the original decimal number much faster than a conventional counter starting from 0 and counting all the way up. SAR ADCs are frequently used for inexpensive implementation with high to medium resolution, typically with sampling rates less than 5MS/s [12]. They are ideal for data/signal acquisition and battery powered instruments due to their low power consumption.

SAR and the flash ADCs are the ADCs used in this thesis.

3.2. Review of floating-point A/D converters

Floating point Analog-to-Digital Converters (FP-ADC) are used for acquiring signal within a high dynamic range while minimizing the relative quantization error [3], [14]. There are many types of analog-to-digital converters: ADCs with automatic gain control (AGC), ADCs with variable gain stages, algorithmic FP-ADCs and so on [16]. Two classic solutions are mostly used to implement FP-ADCs so it is more suitable to classify them into two categories based on their quantization method.

The first category is the ADC with non-uniform quantization where the input signals are quantized in to steps that are not spaced uniformly. In other words, the quantization steps are small for smaller inputs and large for larger inputs which can be seen in Figure 3-1

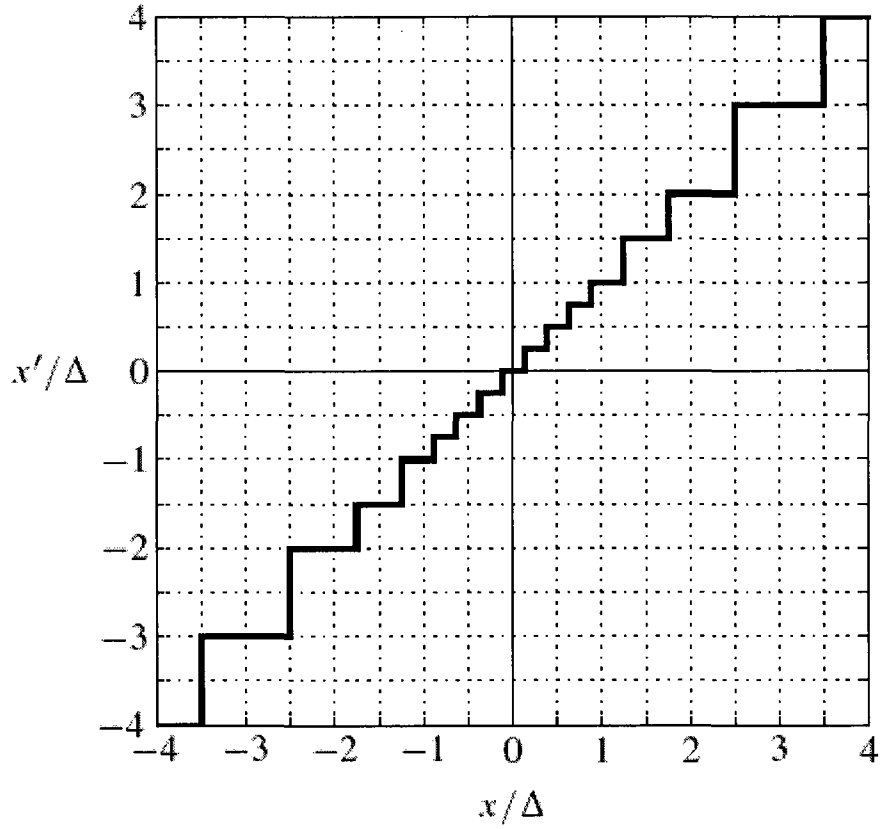

Figure 3-1 A non-uniform floating-point quantizer with a 2-bit mantissa [18]

One way to implement a floating-point ADC would be to use a logarithmic ADC and convert the compressed digital output into a floating-point notation. This method is based on the observation that floating-point numbers can be derived by replacing the logarithm of the digital signal.

Rewriting the equation for the digital value of a floating-point quantization (2-15) as follow:

$$d = d_M 2^{d_E} + \frac{V_{FS}}{2} \tag{3-4}$$

Where

$$d_E = w_E - 2^{n_e} + 1 \tag{3-5}$$

$$d_M = w_M \cdot \frac{V_{FS}}{2^n} \tag{3-6}$$

Then d_M and d_E in equation (3-4) can be solved as in reference [28]:

$$d_E = \left\lfloor \log_2 \left| d - \frac{V_{FS}}{2} \right| \right\rfloor + 1 \tag{3-7}$$

$$d_M = \frac{d}{2^{d_E}} \tag{3-8}$$

Where $\lfloor . \rfloor$ is the floor function.

And according to the c-law (3-1) the logarithm of the signal is:

$$d_L = f_c(d) = \frac{\ln 2}{\ln c} \frac{\ln(|d|)}{\ln 2} + 1 = \frac{\ln 2}{\ln c} \log_2 |d| + 1 \tag{3-9}$$

Finally, by replacing the logarithm of the digital signal in (3-7) and (3-8):

$$d_E = \lfloor d_L \rfloor \tag{3-10}$$

$$d_M = sign(d - V_{FS}/2).2^{d_l - d_E} \tag{3-11}$$

Where the exponent is the integer part of d_L and the mantissa is two to the power of the fractional part of d_L. The mantissa's sign is considered separately.

The non-uniform quantizer method has its share of advantages and disadvantages. As discussed in the problem definition under the introduction section, many applications do not require high precision or resolution at large amplitude so there is no useful advantage at this point from providing a large number of bits to meet a higher accuracy at all levels. The only advantage from this is simply to provide a wide dynamic range [1]. As a consequence, the non-uniform FP-ADC perform well at this level and also leads to smaller quantization power error because the quantizer's input distribution is similar to a normal distribution in this case [17]. However, one major drawback of this method is the

increased hardware complexity due to the required large number of unit used in the layout to match each quantization step. Therefore, if the length of the registers is not sufficient the precision of the A/D converter will suffer and the error will worsen.

The second category includes ADCs with uniform quantization where the quantization steps are uniform across the input range which can be seen in Figure 3-2.

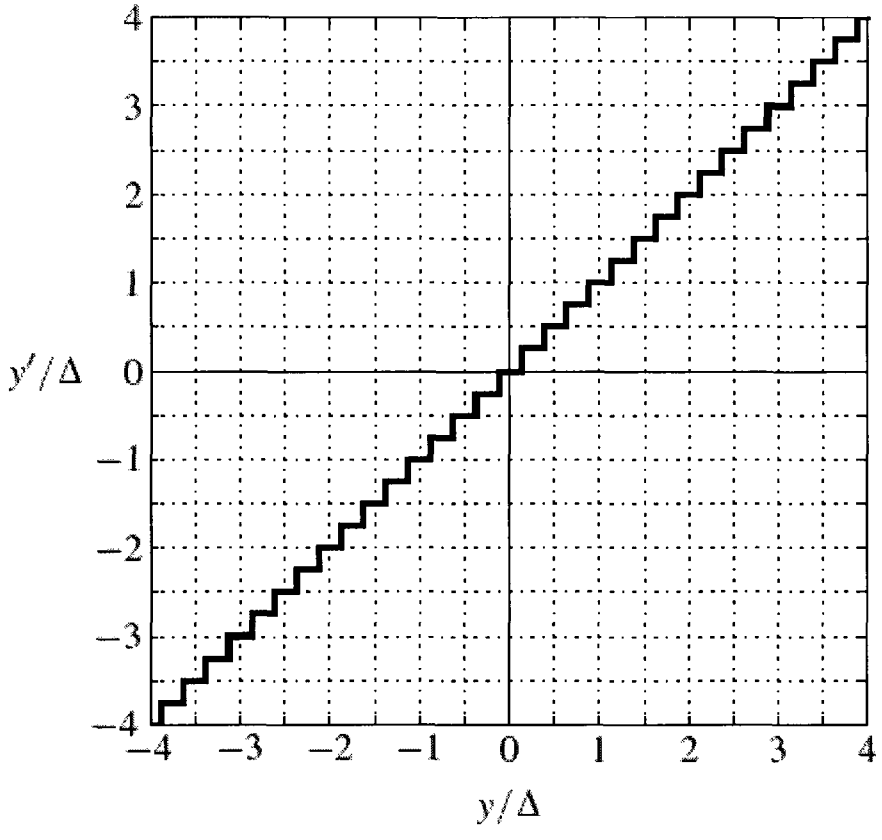

Figure 3-2 A uniform quantizer with a 2-bit mantissa [18]

In comparison to fixed-point Analog-to-digital converter, FP-ADCs have a higher quantization range and are therefore far superior to their ADC counterpart from a statistical point of view [19]. However, the conversion time is doubled with FP-ADC

because they need two stages to calculate the exponent and mantissa and complete the conversion. In this type of configuration called sequential, two-cycle or sometimes 2-step floating-point A/D converter [19] [20], the exponent is first determined in the logarithmic ADC. The exponent controls the gain of an amplifier that normalizes the signal. The amplified signal is converted in the linear ADC which produces the mantissa.

In another variant, both the exponent and mantissa are determined by the same linear ADC [5] [19] [23] [24] [25] [26]. The ADC is connected to the acquired signal through a programmable gain amplifier (PGA) as shown in Figure 3-3.

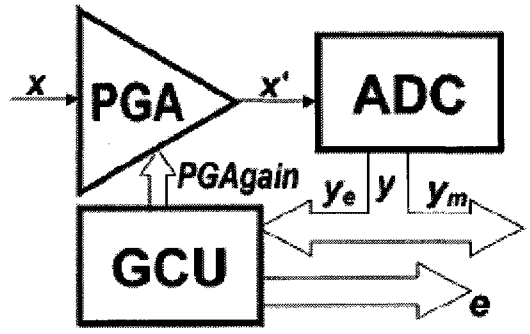

Figure 3-3 Block diagram of sequential floating-point ADC

At the beginning, the ADC performs a conversion cycle to find out the exponent, then set the PGA gain accordingly and performs a second cycle to retrieve the mantissa. This technique uses a variable gain amplifier that varies in the steps by power of 2. Depending on the input voltage range, it attempts to bring the amplified signal in the upper-half of the converter range where it performs better. The gain is initially set to its lowest value and the signal is converted by the linear ADC, the controller determines the exponent from the ADC output and sets the amplifier gain thereafter. A consecutive

conversion is performed by the ADC to resolve the mantissa. While this solution preserves the precision of uniform ADCs and avoids separate coding stages, the effect of using the same ADC requires two conversions for one sample and therefore doubles the conversion time.

One possible solution, described thoroughly in [25], implements what is called a distributed FP-ADC. It overcomes the slow conversion time that characterize the sequential ADC. In the distributed FP-ADC, the mantissa gain is provided by a distributed amplifier. The input signal is amplified by a distributed amplifier exemplified by a chain of amplifiers each having a gain of 2. The outputs of the distributed amplifier which are binary weighted are measured by a logarithmic ADC. The ADC controls the switch to select the appropriate gain and determines the mantissa. The linear ADC converts the mantissa.

Another potential solution to shorten the conversion time is to implement the architecture of a pipelined floating-point ADC as described exhaustively in [25]. In the pipelined ADC, the exponent and the gain of the amplifier are determined rather consecutively in pipeline stages. In the first pipeline stage, the most significant bit (MSB) of the exponent is determined by the logarithmic ADC. If the MSB is set then the signal is amplified by 1 otherwise by $2^{2^{n_e-1}}$. In the next pipeline stage, the second MSB is determined. This way, the exponent and the gain are determined in a binary step fashion until the complete exponent is resolved. The gain has been set when the exponent was resolved so the mantissa can be converted by the linear ADC.

The parallel uniform ADC, as discussed in [1] and [19], further enhances the conversion time of the sequential implementation by overlapping the fine quantization

cycle over the last part of the coarse quantization which at the end speed up the process of conversion.

The proposed circuit presented in depth in [19] [20] [21] and shown in Figure 3-4, consists in two distinct flash ADCs, this time working simultaneously: one determines the exponent, while the other one, connected to the quantizer input over a programmable gain amplifier, finds the mantissa.

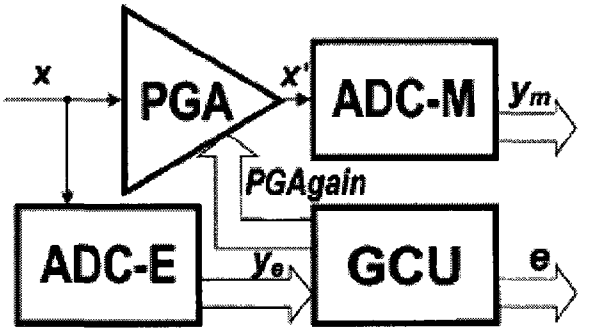

Figure 3-4 Block diagram of parallel floating-point ADC

If two adjacent samples have the same exponent thus the same gain, the conversion result is delivered immediately; otherwise the mantissa is acquired again with the PGA gain reset to the most recent gain based on the history of the acquired signal and the conversion process is complete. In this later case there is no improvement over the sequential ADC as it takes two cycles for the process to finish. In the first case however, the conversion time is halved. As a consequence this architecture relies mostly on the goodness of the predictive algorithm responsible on setting the gain for the PGA when calculating the mantissa. The simplest approach of a zero degree polynomial extrapolation is considered in [19] to illustrate the gain prediction of the parallel floating-point ADC.

It is important to point out though, that another category of quantization emerged and is called semi-uniform quantization which is simply speaking a combination of uniform and non-uniform quantization with the middle $(2^{k-1}-1)$ quantization steps being the same as those of a (k+1)-bit quantizer, while the other 2^{k-1} steps are three times as large. The semi-uniform quantization is said to have the advantages of both quantizers mentioned. That is the simple structure of the uniform quantizer and the high dynamic range of the non-uniform one [22].

Chapter 4

4. System Description

This chapter presents the architecture of the system being investigated by this thesis that is the sequential architecture of the floating-point A/D converter. It also covers the different interactions between the ALTERA board and the daughter card on which the FP-ADC is implemented.

4.1. The proposed FP-ADC architecture

Based on a uniform quantizer that preserves the precision of the sequential floating-point A/D converter, while minimizing the conversion time as much as possible, a new architecture of the floating-point analog-to-digital converter is proposed. This 2-step architecture achieves a high throughput rate by combining two sample-and-hold (SH) amplifiers and a fast A/D converter. The sampling interface is optimized by overlapping the acquisition time of the first sample-and-hold amplifier and the settling time of the gain amplifier with the conversion time of the A/D converter. The second SH amplifier holds the amplified signal while the A/D converter perform its conversion routine. The proposed solution takes advantage of two A/D converters that work in tandem; one determines the exponent (A/D-E), while the other one which is connected to the quantizer input over a programmable gain amplifier (PGA) finds the mantissa (A/D-M). The PGA is software based and is connected to the exponent output. The block diagram of the floating-point A/D converter is shown in Figure 4-1.

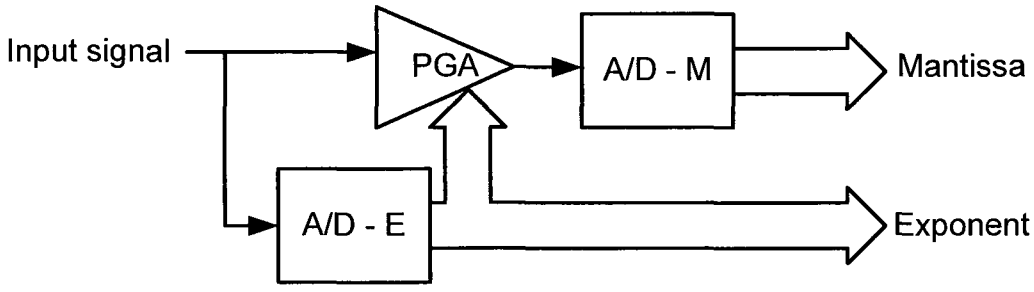

Figure 4-1 Block diagram of sequential floating-point A/D converter

The Figure bellow shows the flowchart of a floating-point A/D conversion. It produces two parts of digital output data consisting of an exponent and a mantissa.

The first operation is to detect the level, or range of the input signal magnitude. This range is encoded as an m-bit exponent. Next, the signal is quantized uniformly to obtain an n-bit mantissa. The step size with which the mantissa is quantized is scaled according to the exponent. The total resolution of the converter is n + m bits [29].

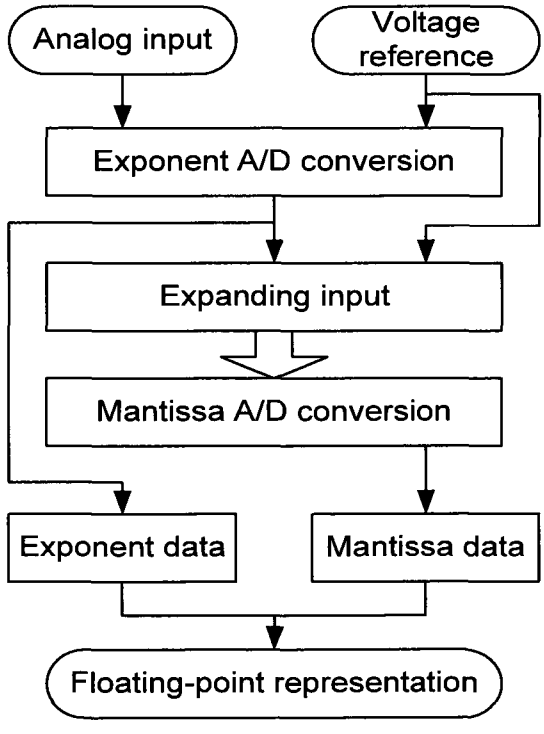

Figure 4-2 Flowchart of a floating-point A/D conversion

4.2. System Architecture

The dichotomy of the system shows two major components:

- A printed circuit board/ daughter card which implements the FP-ADC. It consists of two A/D converters, the A/D-E and the A/D-M converters that get the exponent and the mantissa respectively; a programmable gain amplifier which is a digitally controlled gain amplifier with binary model and a pair of sample-and-hold amplifiers which determines the flow control.
- The FPGA (Altera Stratix), which implements the serial communication interface between the sequential architecture of the FP-ADC and a PC.

The main architecture of the system is shown in Figure 4-3

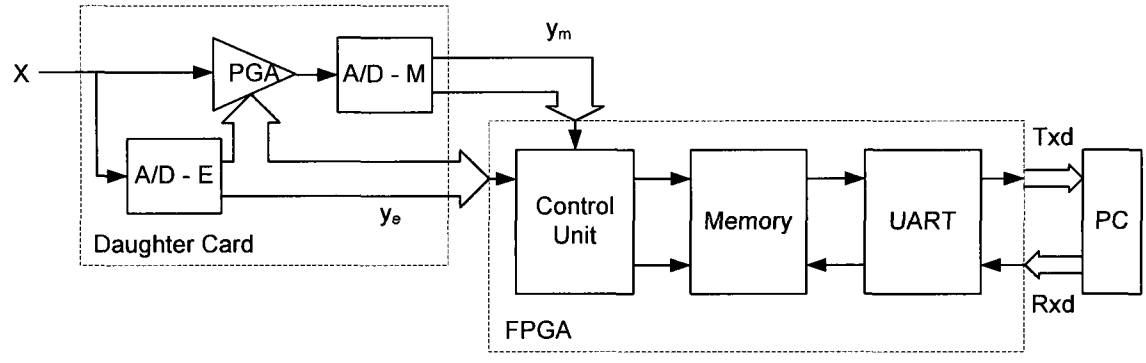

Figure 4-3 Block diagram of the system architecture

When the FP-ADC is ready to begin a floating-point analog-to-digital conversion, a "start" command is sent from the PC as an encoded message "RxD" (data reception) via the RS-232 port to the FPGA UART at a maximum baud rate of 115200 Bps. The Control Unit decodes this message and starts the floating-point A/D acquisition. Every floating-point result is sent in a DMA (Direct Memory Access) manner to a buffer that is implemented as a dual-port memory block. When the number of the sample data written

reaches the maximum allowable space in the memory, the floating-point A/D conversion stops and the acquired data (the mantissa data and the computed exponent) is read and sent by the FPGA serially as "TxD" to the PC through the UART for further processing. The FP-ADC subsystem is discussed in the following.

4.3. Daughter Card FP-ADC Subsystem Description

The floating-point analog-to-digital converter is based on reference [31]. It is diagrammed in Figure 4-4. It consists of a pair of sample and hold amplifiers (the AD585), a flash converter, a five-range programmable gain amplifier (the AD526) and a fast successive approximation register (SAR) 12-bit A/D converter (the AD7572).

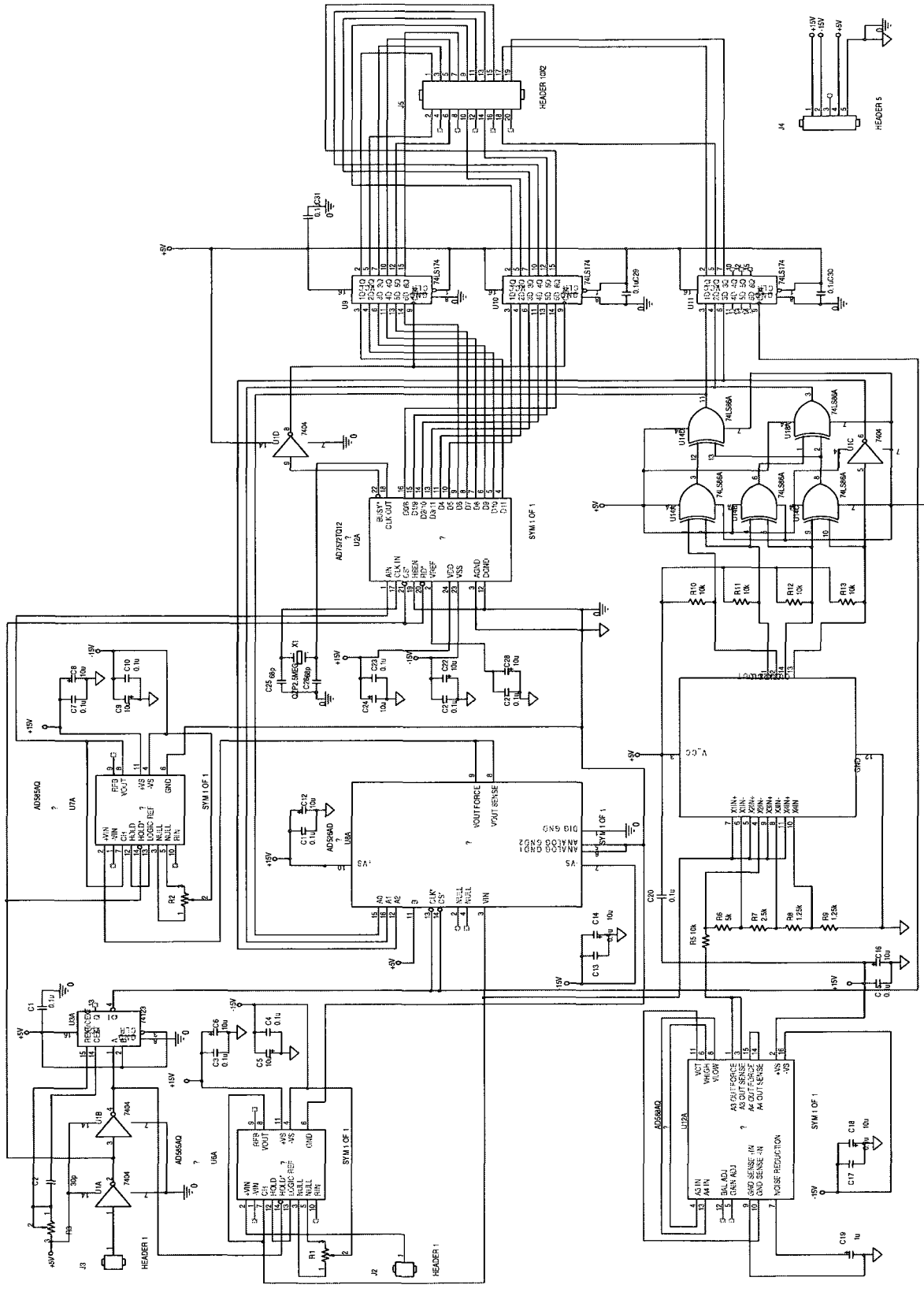

Figure 4-4 FP-ADC Schematic

The output data is presented as a 16-bit word; the lower 12 bits from the A/D converter form the mantissa and the upper 4 bits from the digital signal used to set the gain form the exponent. The AD526 programmable gain amplifier in conjunction with the comparator circuit scales the input signal to a range between half-scale and full-scale for the maximum usable resolution. This ensures that the signal is converted effectively by the A/D converter.

The floating-point analog-to-digital converter achieves its high throughput rate of 125 KHz by overlapping the acquisition time of the first sample and hold amplifier and the settling time of the AD526 with the conversion time of the A/D converter. The first sample and hold amplifier holds the signal for the flash auto-ranger which consists of four comparators (the LM339) connected in parallel with reference voltages set by a resistor network that determines which binary quantum the input fall within relative to full-scale. This is known as thermometer code encoding in analogy to the mercury column that always rises to the appropriate temperature and no mercury is present above that temperature. So it is similar to the point where the code changes from ones to zeros when the input signal becomes smaller than the respective comparator reference voltage levels. The outputs of the comparators O_1, O_2, O_{13} and O_{14} are connected to the gain amplifier inputs A_0, A_1 and A_2 through logic gates where:

$$A_0 = (O_{13} \oplus O_{14}) \oplus (O_1 \oplus O_2) \tag{4-1}$$

$$A_1 = (O_{13} \oplus O_{14}) \oplus (O_1 \oplus O_{14}) \tag{4-2}$$

$$A_2 = \overline{O_{13}} \tag{4-3}$$

Five levels of gain could be set; 16, 8, 4, 2 or 1(no gain). These are depicted in Figure 4-5 for a triangular input signal. It can be seen that the signal is amplified so that it stays in the upper-scale (2.5V to 5V) of the input range.

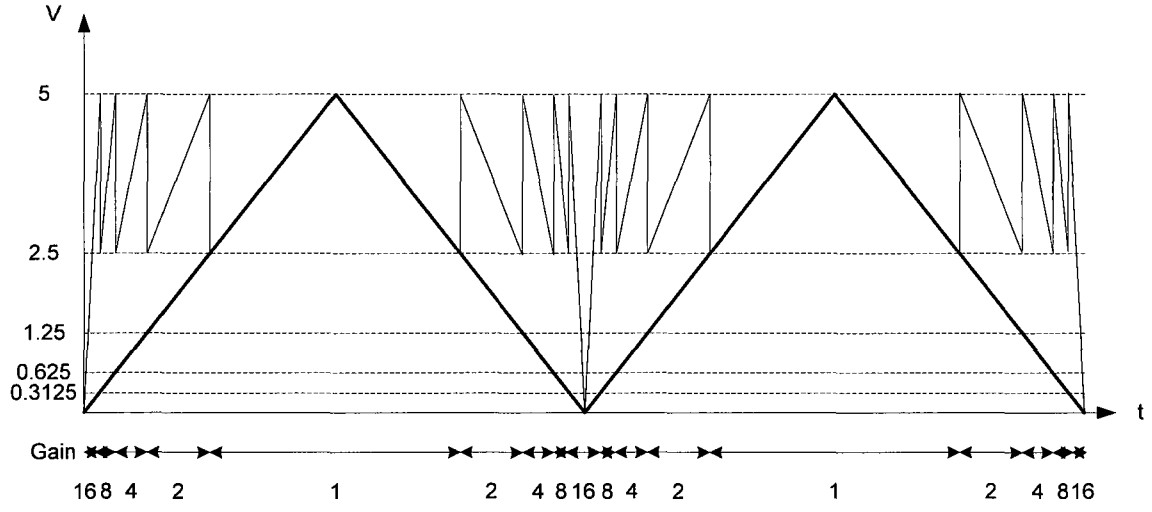

Figure 4-5 Gain settings for a triangular input signal

The relationship between the gain and the exponent is summarized in the following table.

Table 4-1 Gain and exponent relationship

Signal	Exponent	Gain
5 – 2.5 V	000	$1 = 2^0$
2.5 – 1.25 V	001	$2 = 2^1$
1.25 – 0.625 V	010	$4 = 2^2$
0.625 – 0.3125 V	011	$8 = 2^3$
0.3125 – 0 V	111	$16 = 2^4$

At this time, once the AD526 gain is set and it has settled to the appropriate level, then the second sample and hold amplifier can be put into hold which holds the amplified signal while the AD7572 perform its conversion routine.

The sample and hold (S/H) circuit is the key element in the data acquisition and A/D conversion processes. It has two basic and distinct operational states. In one state (sample) it samples the input signal and transmits it to the output simultaneously. In the second (hold), it holds the last value sampled until the input is sampled again. Therefore the S/H circuit is necessary for ADC front-end circuits as it maintains at a constant level the input signal to allow the ADC to convert the voltage to a corresponding digital word.

The Figure 4-6 bellow shows the S/H waveforms with the input being sampled at the top, the S/H control at the middle and the S/H output at the bottom.

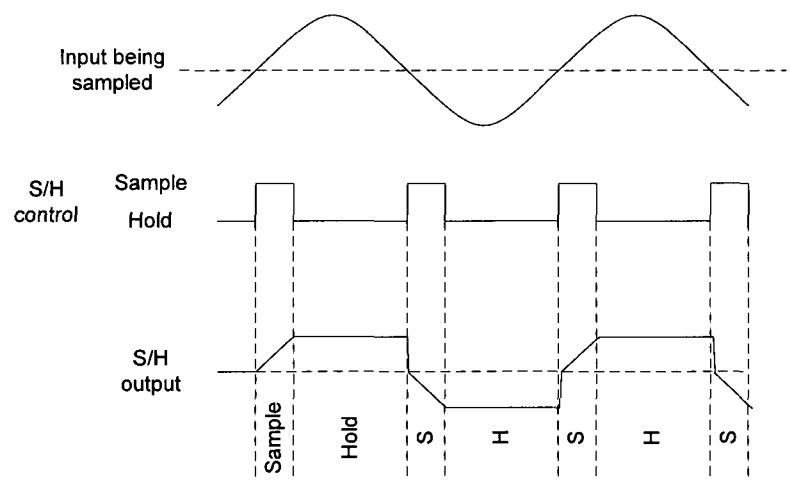

Figure 4-6 S/H Waveforms

According to the specifications of each component in the FP-ADC in Figure 4-4, the sample and hold amplifier AD585 acquisition time is 3 μs. The conversion time of the A/D converter is 5 μs. So in total it takes 8 μs or 125 KHz. This performance is

achievable thanks to the fast settling time of the programmable gain amplifier AD526 once the flash converter (comparator circuit) quantized the input signal. A series of registers holds the 3 bits output from the flash auto-ranger and the 12 bits output from the A/D converter.

The schematic is then translated to generate the layout of the daughter-card depicted in Figure 4-7

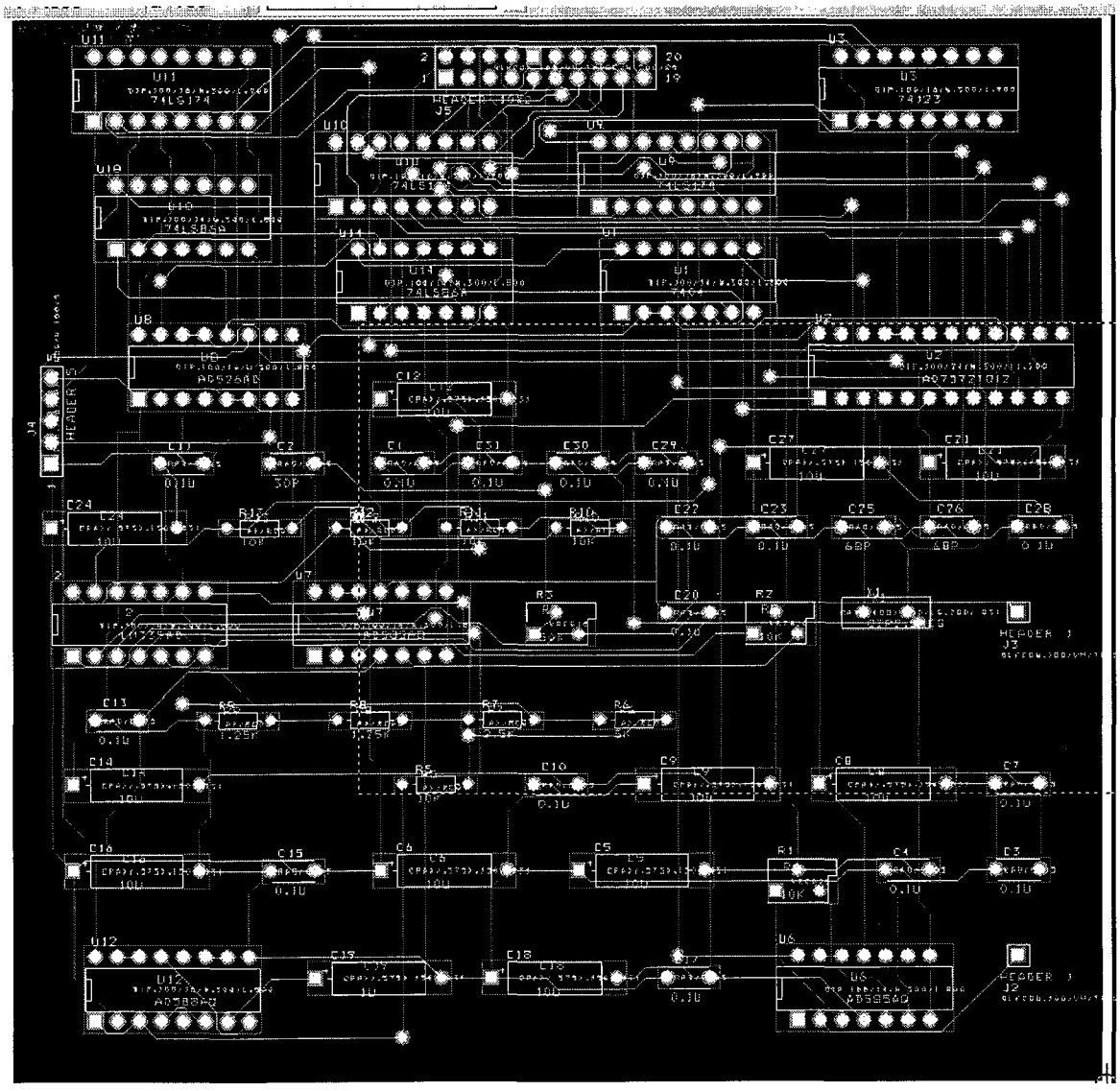

Figure 4-7 FP-ADC Layout

The last process is the manufacturing of the PCB. The daughter-card picture is shown in Figure 4-8.

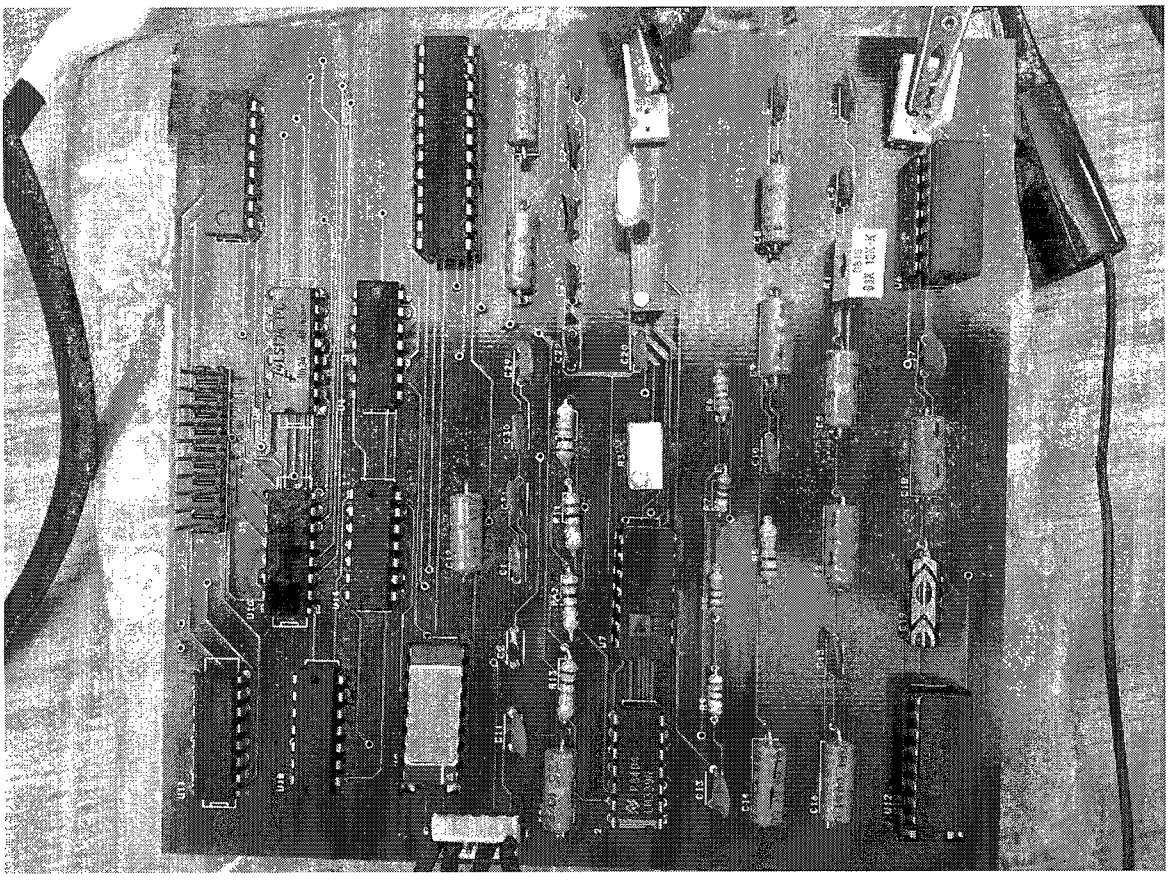

Figure 4-8 FP-ADC Daughter-Card

4.4. FPGA Subsystem Description

The FPGA subsystem is implemented on an Altera Stratix EP1S80B956C6 DSP development board. The board has A/D converters, D/A converters, SRAM and Flash memories, switches, LEDs, RS-232 connector, expansion interfaces and on board 80-MHz oscillator [30].

From Figure 4-3, it is easily seen that the FPGA subsystem consists of three main modules:

- The UART module which is responsible for the data communication between the FPGA and the PC.
- The memory module which acts as a buffer where the data is stored.
- The control unit which is the link that controls the data flow between the daughter card, the FPGA and the PC.

VHDL is the programming language used for the implementation of these different modules and RTL (Register Transfer Level) is used for the design process. The main architecture of the FPGA subsystem is reproduced in more detail in Figure 4-9.

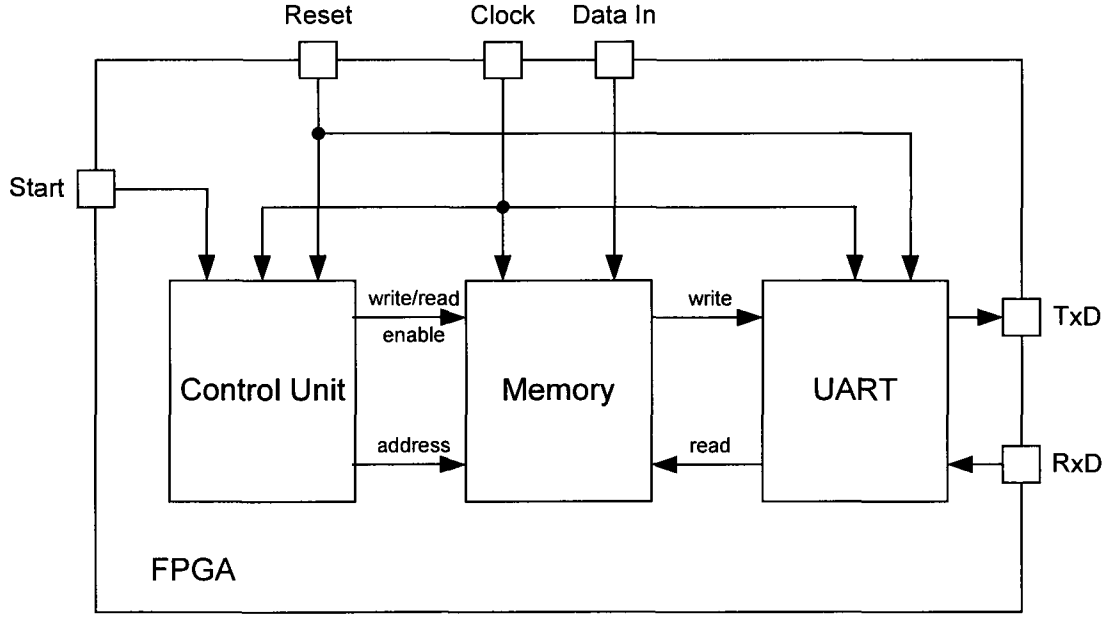

Figure 4-9 FPGA subsystem

In VHDL design, TestUART.vhd is the top-level hierarchy module of the FPGA subsystem. All sub-modules are instantiated here. There is no logic present in this

module. It consists of 12 sub-modules classified according to three main functions: UART, Memory and Control Unit which provide the mean to initialize the data acquisition, get the exponent and mantissa, store them on to the memory, fetch the information from the memory and send it serially to the PC.

The Control Unit is the entity responsible for starting the acquisition of the data and coordinating the flow of information read to the memory and written from it. It is also in charge of the data being displayed by the seven segment display and the status LEDs. It consists of the following VHDL entities: clokDivider.vhd. seg7.vhd, enARdFF_2.vhd, sevenBitRegister, FsmController.vhd and Start.vhd.

The Memory block manages the data stored on the FPGA. This data is the exponent and mantissa being sent by the FP-ADC on the daughter card. When the sample data reaches the limit which is the maximum size of the RAM, the writing process is interrupted and the floating-point conversion stops so that the data is fetched and sent to the PC through the UART. The memory consists of the following VHDL entities: FsmController.vhd, RAM.vhd, controller.vhd and data.vhd.

The UART refers to the logic controller of the UART implemented on the FPGA. The input data can be sent from a PC's serial COM port to the UART pin RxD in the Stratix board via a RS-232 cable. The PC can receive the output data from the Memory module using pin TxD on the FPGA and send back the control command through the UART as well. The UART is a universal asynchronous receiver and transmitter core in VHDL. It consists of the following components: recepteur.vhd, emetteur.vhd, baud.vhd which will be discussed in further detail.

The system can be reset at any time by asserting the signal RESET by pressing the master reset switch SW0. This pin is used as a standard I/O pin to implement a reset in the design.

The table bellow summarizes the default port list of the TestUART.vhd top-level entity. In this table, signals with prefix "i" means input and "o" means output.

Table 4-2 Default ports list of TestUART.vhd

Signal	Width	Description
i_resetb	1	Global rest for the system
i_clock	1	Main system clock
i_RxD	1	Received serial data
i_start	1	Start signal
i_data	16	A/D converter data
o_RRDP	1	Status for the receiver
o_7seg1,o_7seg2	7	Output for the seven segment right and left
o_RTDV	1	Status for the transmitter
o_TxD	1	Transmitted serial data

The UART is a universal asynchronous receiver and transmitter core that can work fully functionally and synthetically. The receiver and transmitter are two independent modules that provide the communication between two serially connected devices. The following modules were designed, implemented and tested.

The Table 4-3 summarizes the default port list of the transmitter module emetteur.vhd.

Table 4-3 Default ports list of emetteur.vhd

Signal	Width	Description
i_resetb	1	Global rest for the system
i_BClk	1	Baud generated clock
i_RTDV	1	Input status: data transmission register empty
i_emetteur	8	Data to be transmitted
o_RTDV	1	Output status: data transmission register empty
o_TxD	1	Transmitted serial data

The Table 4-4 summarizes the default port list of the receiver module recepteur.vhd.

Table 4-4 Default ports list of recepteur.vhd

Signal	Width	Description
i_resetb	1	Global rest for the system
i_BClkx8	1	8 times faster baud generated clock
i_RxD	1	Input received data
o_RRDP	1	Output status: data reception register full
o_recepteur	1	Received parallel data

The next Table 4-5 summarizes the default port list of the baud rate generator module baud.vhd.

Table 4-5 Default ports list of baud.vhd

Signal	Width	Description
i_resetb	1	Global rest for the system
i_clock	1	Global clock for the system
i_Sel	3	Selector for the baud rate
o_BClk	1	Baud generated clock
o_BClkx8	1	8 times faster baud generated clock

4.5. The memory block

The memory consists of the following VHDL entities: FsmController.vhd, RAM.vhd, controller.vhd and data.vhd as depicted in Figure 4-10.

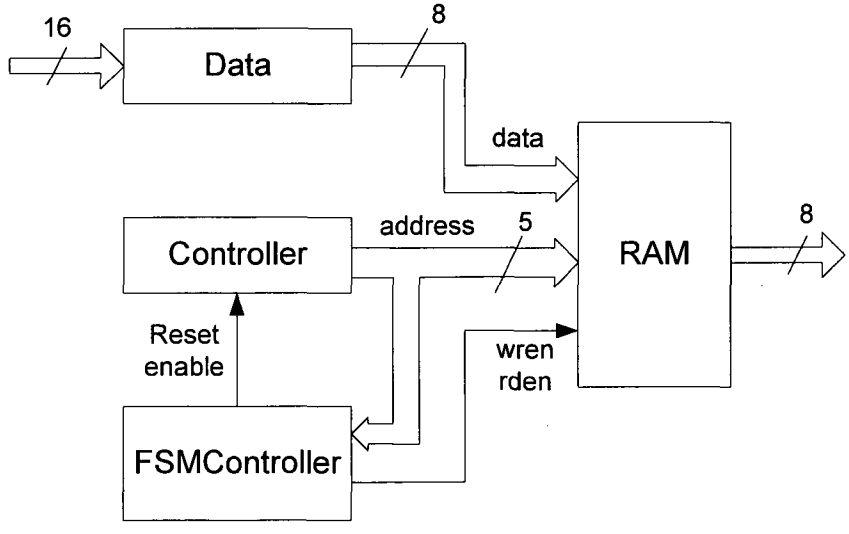

Figure 4-10 Memory block

The RAM module is instantiated by the parameterized dual-port RAM megafunction: altsyncram. The data input bus of the memory is 8 bits wide. The address bus is 5 bits wide. Thus the total memory is 8 bits x 32 words = 256 RAM bits. Note that the size of the memory is expandable and that the choice of this size has been dictated by simplifications made during the design testing procedure. The memory block as generated by the Megafunction wizard within Altera Quartus II IDE is shown in the figure bellow.

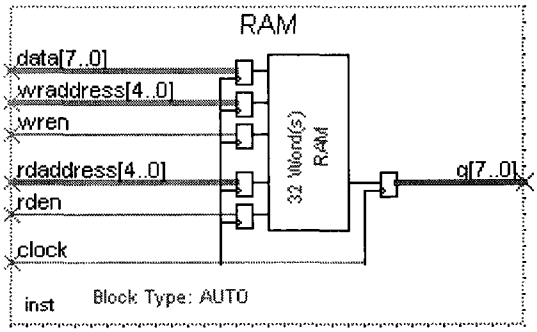

Figure 4-11 RAM module

The writing input ports: "data", "wraddress" and "wren" are registered. The reading input ports: "rdaddress" and "rden" are also registered. A read-during-write output choice when "wren" and "rden" are both set is treated as don't care mode. In this mode, the RAM outputs reflect "don't care".

The controller: controller.vhd is a 5 bits counter that feeds the address ports of the RAM. This module uses reusable coding techniques so that any changes to the size of the memory and therefore the width of the address ports is easily corrected within the instantiation of the module by changing the width parameter (n).

The entity data.vhd splits the incoming data from the FP-ADC from 16 bits to two chunks of 8 bits as illustrated in Figure 4-12. This is due to the data buses inside the

FPGA module being 8 bits wide. This entity makes it easy to maintain the portability of the FPGA module as it is independent from the FP-ADC module.

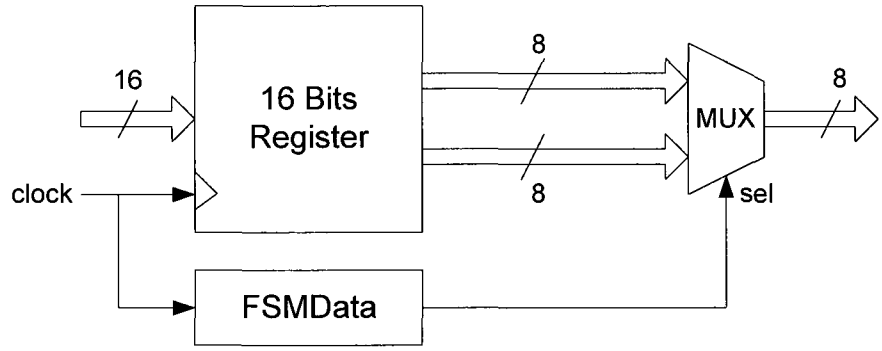

Figure 4-12 Block diagram of data.vhd

The FSMData (Figure 4-13), is a simple two state FSM that controls the MUX selector. The data is outputted sequentially in a block of 8 bits starting by the most significant byte.

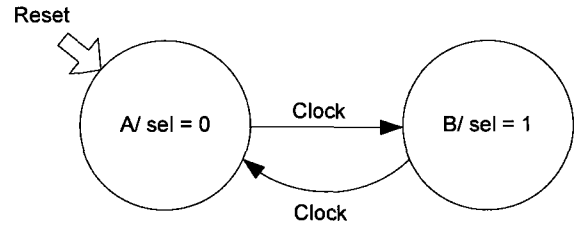

Figure 4-13 FSMData.vhd

Lastly, for the memory block, the FsmController.vhd sends the write and read controls to the RAM. The finite state machine is described bellow.

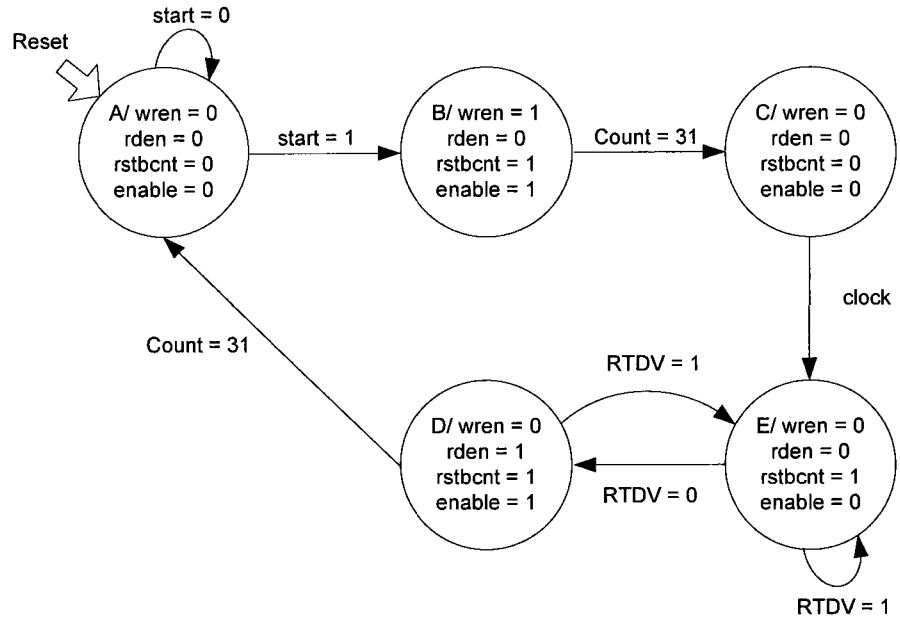

Figure 4-14 FSMController.vhd

Upon reset, the state machine defaults on the idle state A. The state machine is kept on this state as long as the start signal is "0". When the start signal is received, the state machine moves to the writing state B. In this state wren is enabled which means the writing process on the RAM is started. The counter is also started and the writing continues until it runs out of memory space when it reaches the last index indicated by the counter = 31. At this stage, the state machine transitions to the next state to reset the counter and the control signals then moves to state E. The state machine is kept at this state where it waits for the data transmission register. When the data transmission register is ready to receive the new data, the content of the memory is read and ready to be transmitted in state D. Finally, when all of the content of the memory is read the state machine return to the idle state A.

The next table summarizes the default port list of the FSM controller module FSMController.vhd.

Table 4-6 Default ports list of FSMController.vhd

Signal	Width	Description
i_resetb	1	Global rest for the system
i_clock	1	Global clock for the system
i_start	1	Start signal
i_RTDV	1	Data transmission register empty
i_compte	5	Counter for the RAM address
o_wren	1	Write enable for the RAM
o_rden	1	Read enable for the RAM
o_rstbcnt	1	Reset for the counter
o_enable	1	Enable for the counter

4.6. The Control Unit

The Control Unit in Figure 4-15, consists of the following VHDL entities: clokDivider.vhd. seg7.vhd, enARdFF_2.vhd, sevenBitRegister, FsmController.vhd and Start.vhd.

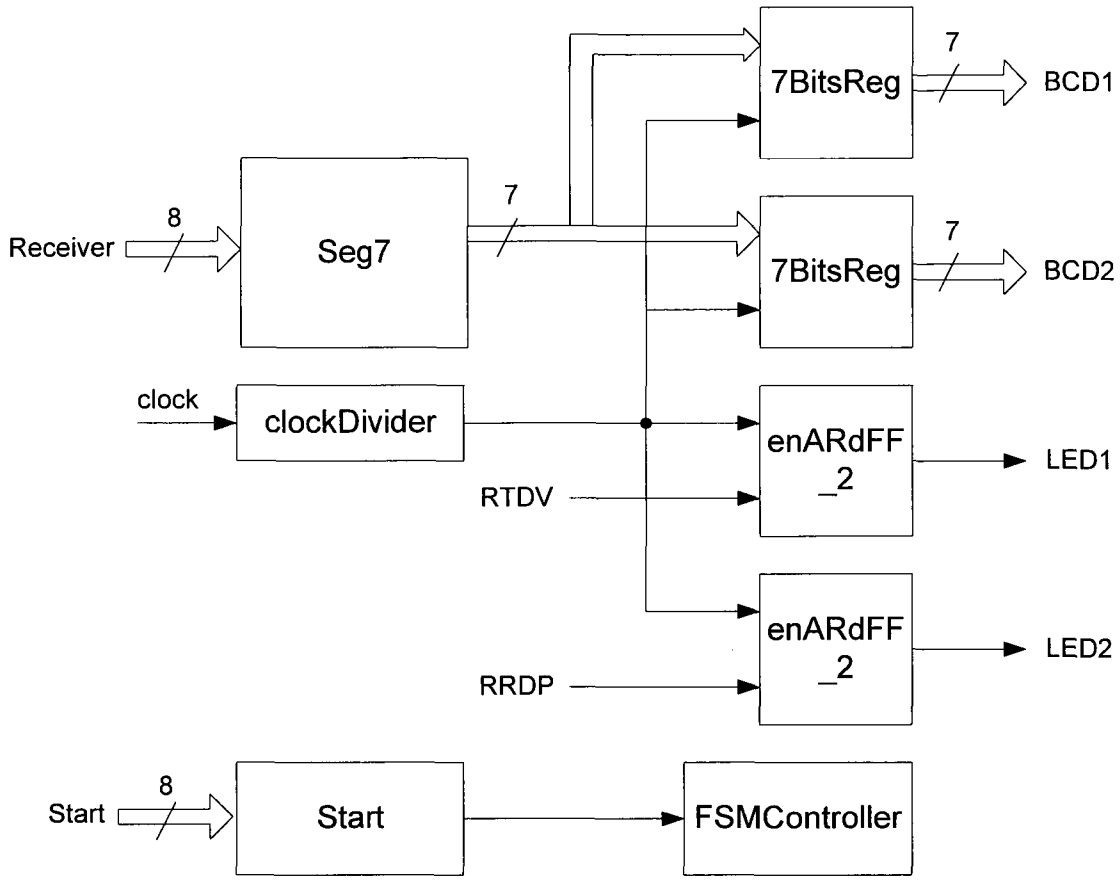

Figure 4-15 Control Unit

The FSMController is already covered in the preceding section. It is connected to the start module which sends the start command "1" upon receiving the character S ("01010011") from the PC.

The seven segment decoder (Seg7) decodes the received character to be displayed on to the seven segments display. Two Binary Coded Decimal (BCD) to seven segments displays are needed in order to output the 8 bits. A clock divider is used to slow the speed of the displayed data.

Finally, for the control unit, the LEDs output the status signals RTDV and RRDP used by the UART.

Chapter 5

5. Simulations and Testing

This chapter presents the simulations and testing of the FPGA and the FP-ADC subsystems. It also discusses the design choices based on the results.

5.1. FP-ADC Subsystem Simulation and Testing

The FP-ADC simulation uses Multisim National Instruments schematic and simulation software. The simulation circuit is shown in Figure 5-1. Some abstractions were necessary during the simulation as some components' model were missing and not available at the moment. Other simplifications were also taken into consideration to ease the work and keep the operation of the circuit understandable within the software limitations.

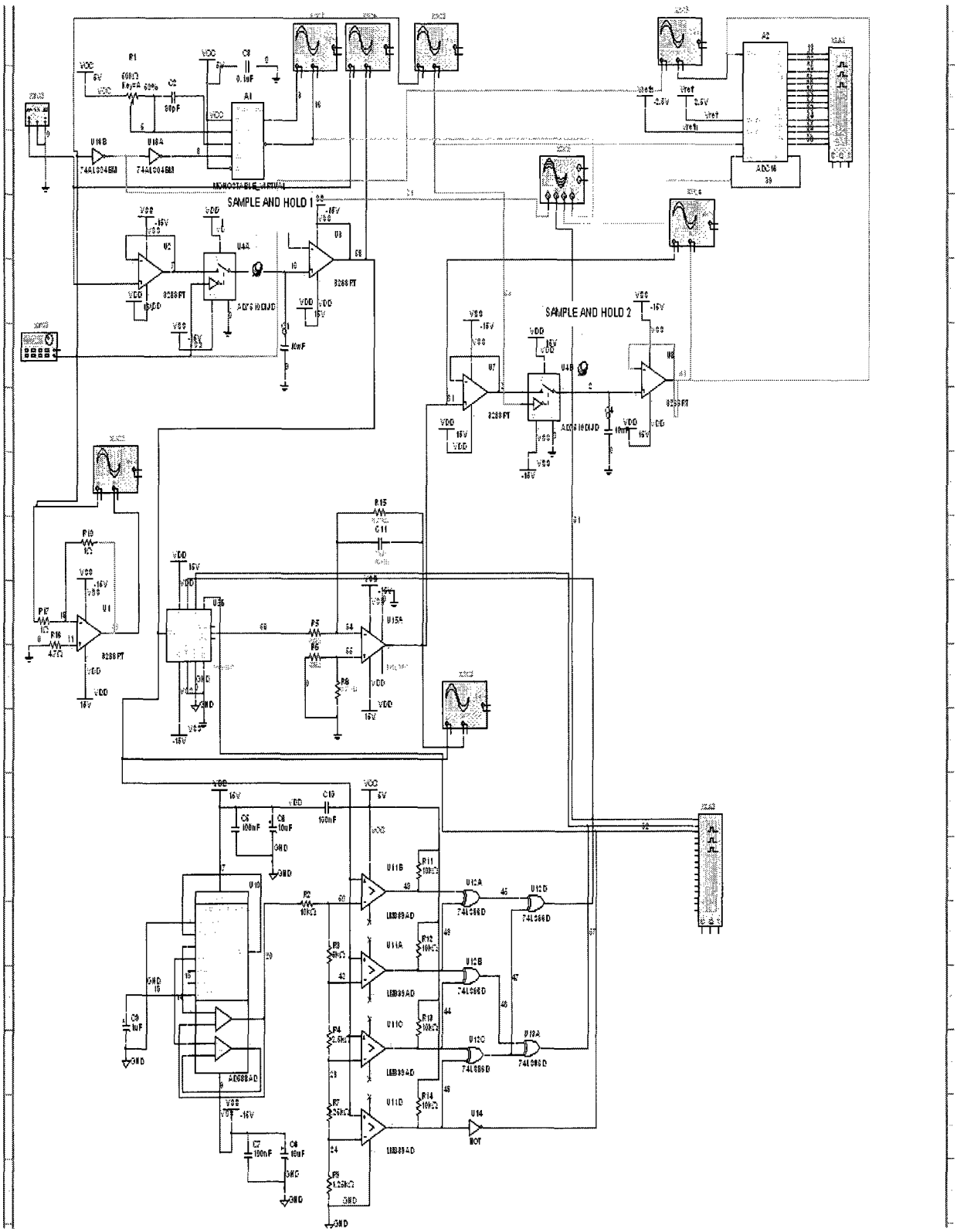

Figure 5-1 FP-ADC Simulation Circuit

The input signal V_{in} at the first sample-and-hold amplifier (SH) is a triangular signal with an amplitude of 1.25V peak-to-peak and a frequency of 1 KHz. The clock signal is a square signal with an amplitude of 5Vpp and a frequency of 10 KHz.

The output of the SH as it tracks the input signal is depicted in Figure 5-2.

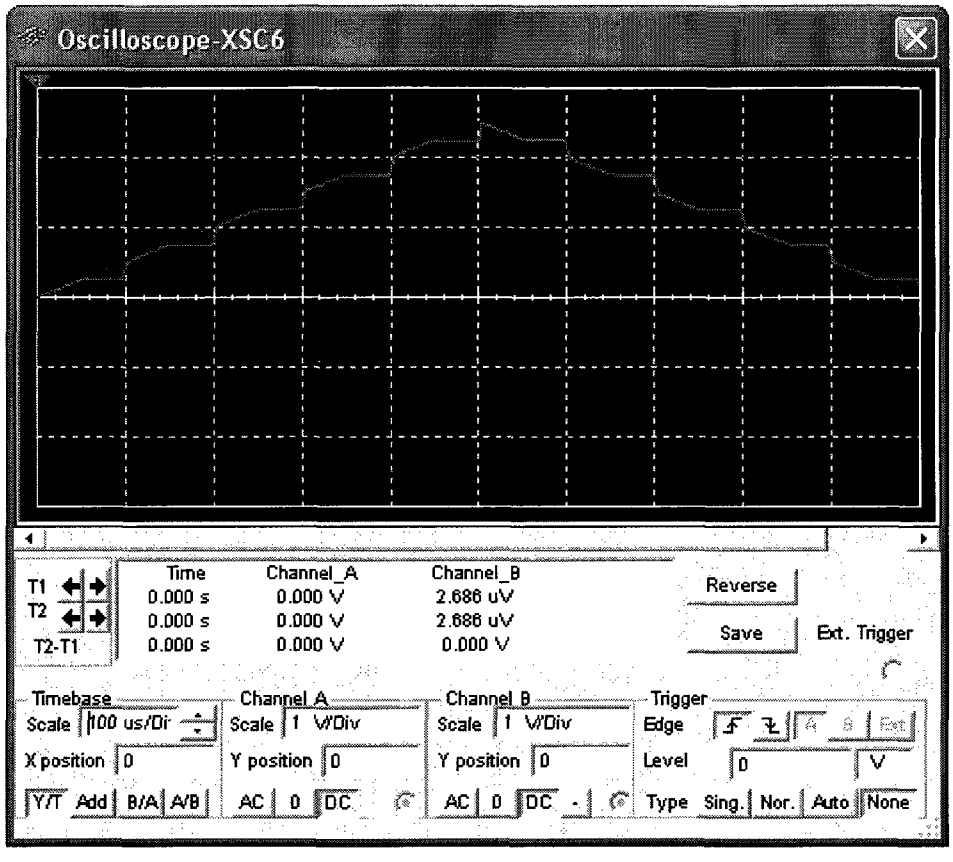

Figure 5-2 SH1 tracking the incoming signal

The output of the SH is then connected to the programmable gain amplifier (PGA). The next simulation shows the output of the PGA. It can be seen that the signal is amplified so that it stays in the upper-scale of the input range.

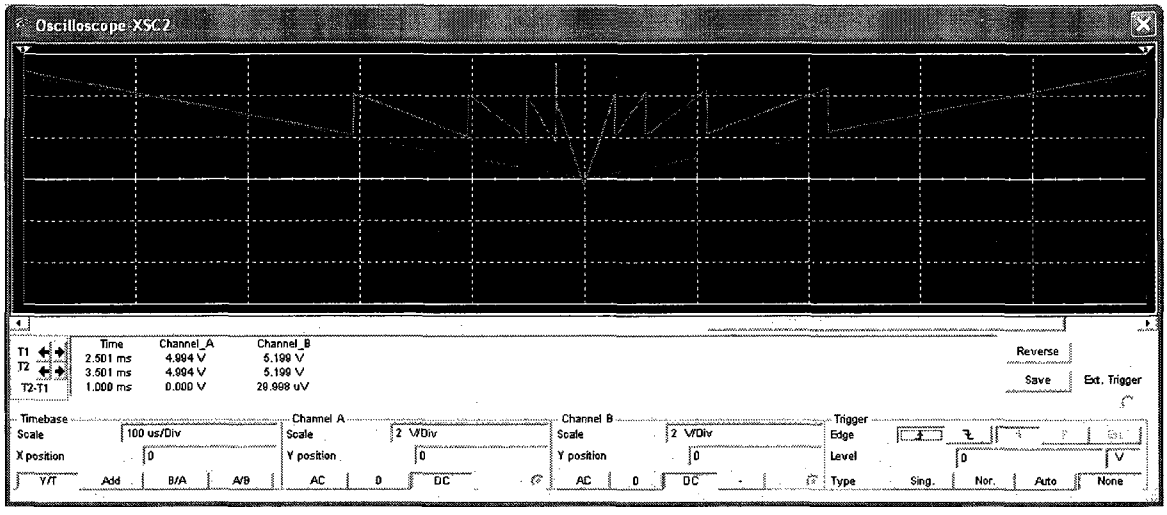

Figure 5-3 PGA Output

The following simulation shows the exponent on the logic analyzer. The output of the flash auto ranger is fed to the PGA and the gain is set accordingly. The relationship between the gain and the exponent can be drawn by superimposing Figure 5-4 and Figure 5-3. It can be easily seen that the gain is initially set to 1 then 2, 4, 8 and 16 and the output result amplified in a way it stays on the upper-scale of the input range.

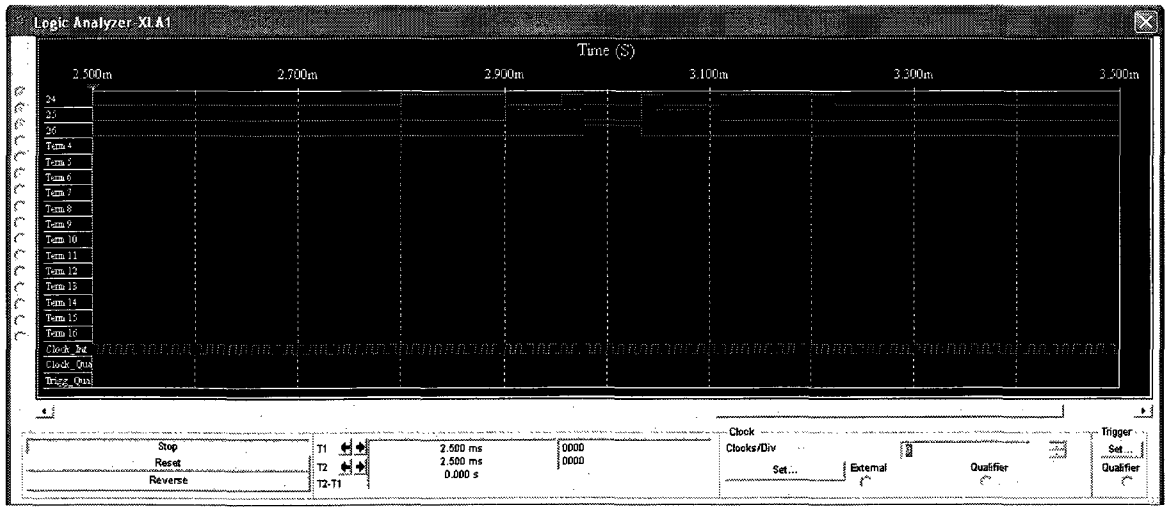

Figure 5-4 The exponent simulation

The next graph depicts the clock signal at the bottom, the monostable multivibrator output in the middle and the outputs of the first SH compared to the second SH (the smoother line at the top).

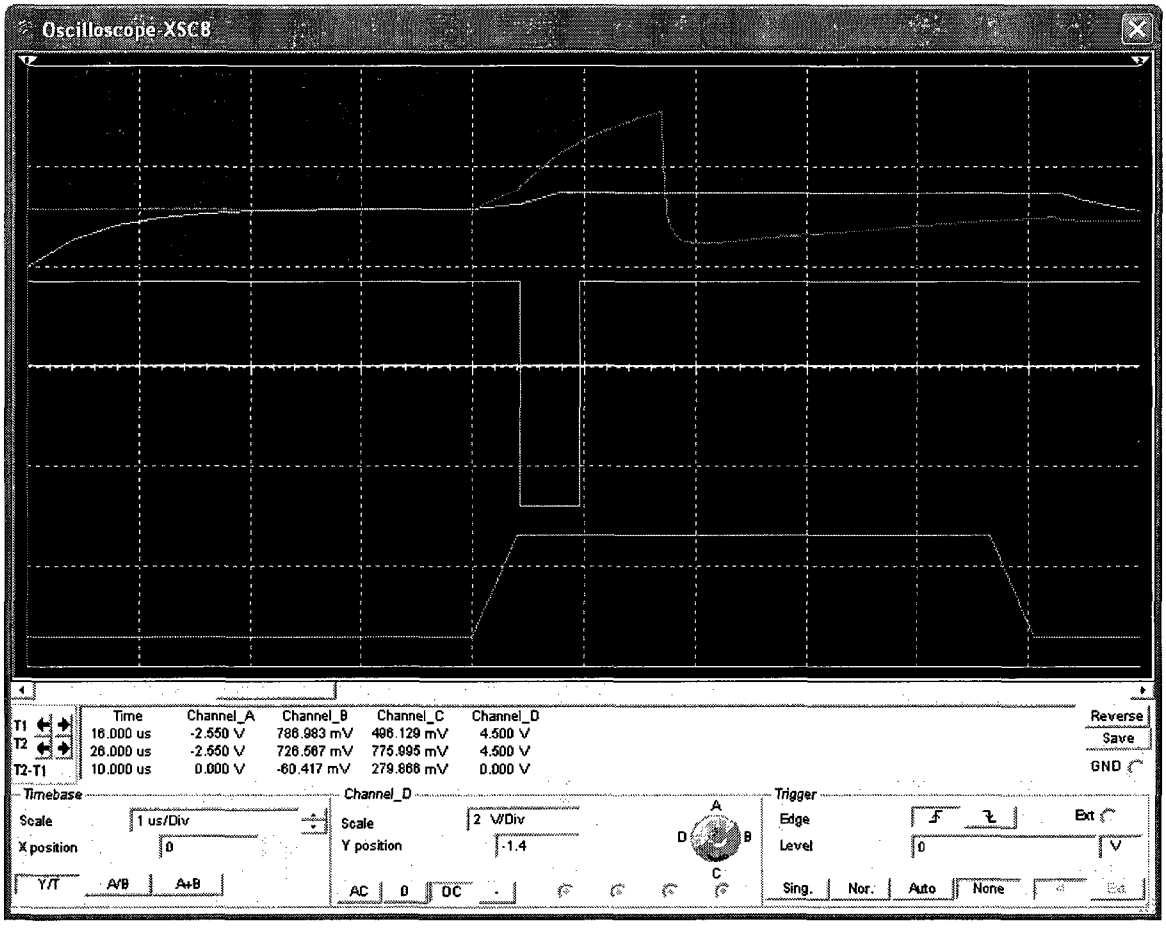

Figure 5-5 SH1 output compared to SH2 output

The next simulation similarly shows the clock signal at the bottom, the monostable multivibrator output in the middle but also the first bit of the gain A0 and the output of SH2 at the top which clearly illustrates the signal being amplified when the gain is $2 = 2^1$.

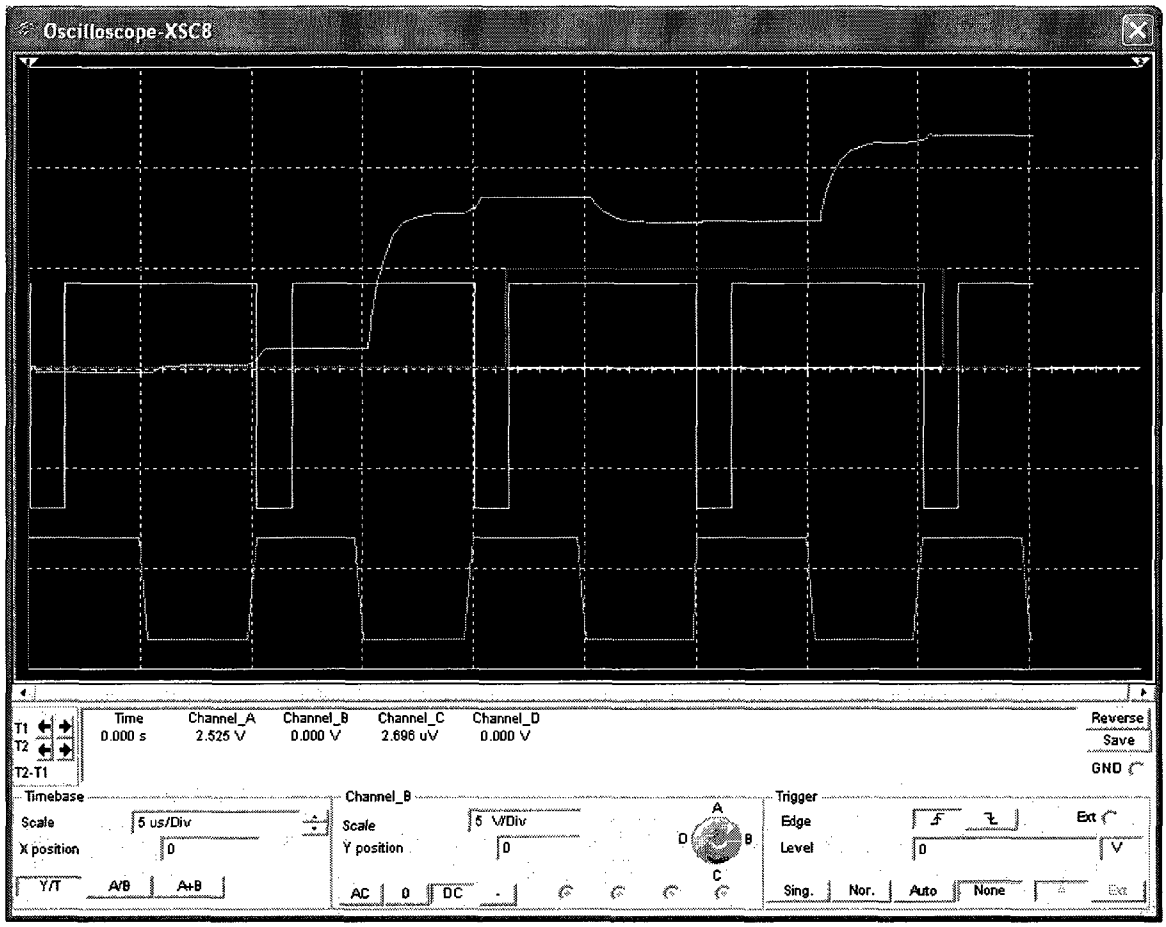

Figure 5-6 SH2 output

The last simulation shows the mantissa outputted by the A/D-M

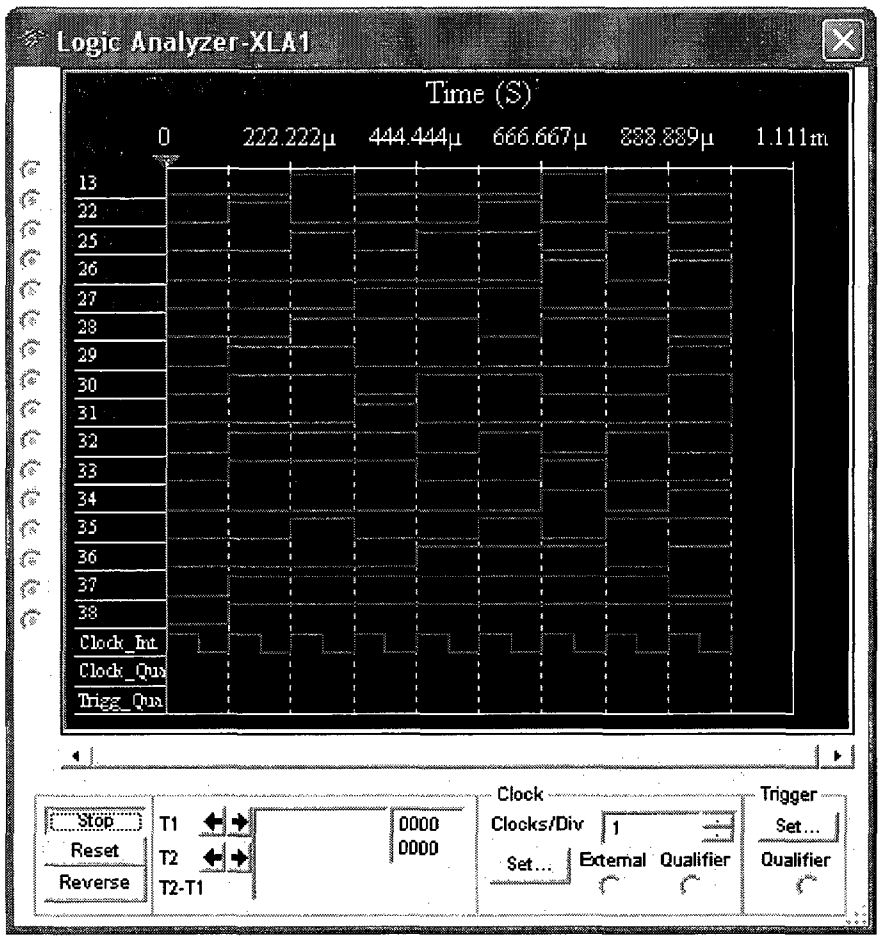

Figure 5-7 The mantissa simulation

For testing, in Figure 4-4 the input signal V_{in} at the first sample-and-hold amplifier (SH) is a triangular signal with an amplitude of 1.25V peak-to-peak and a frequency of 1 KHz (Figure 5-8). The clock signal is a square signal with an amplitude of 2.1Vpp, an offset of 1.2V and a frequency of 125 KHz (Figure 5-9).

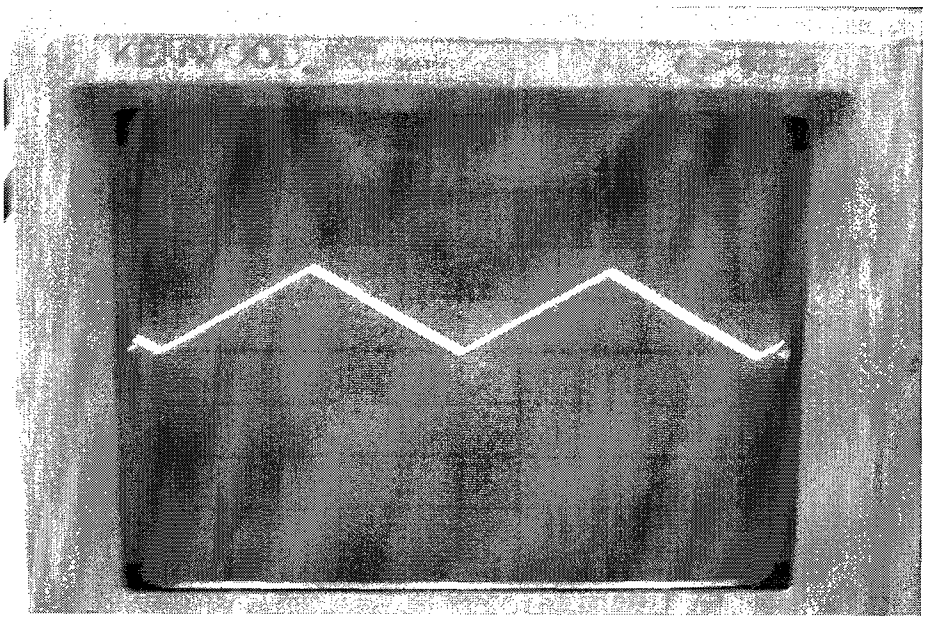

Figure 5-8 Input Signal (1.25Vpp, 1KHz)

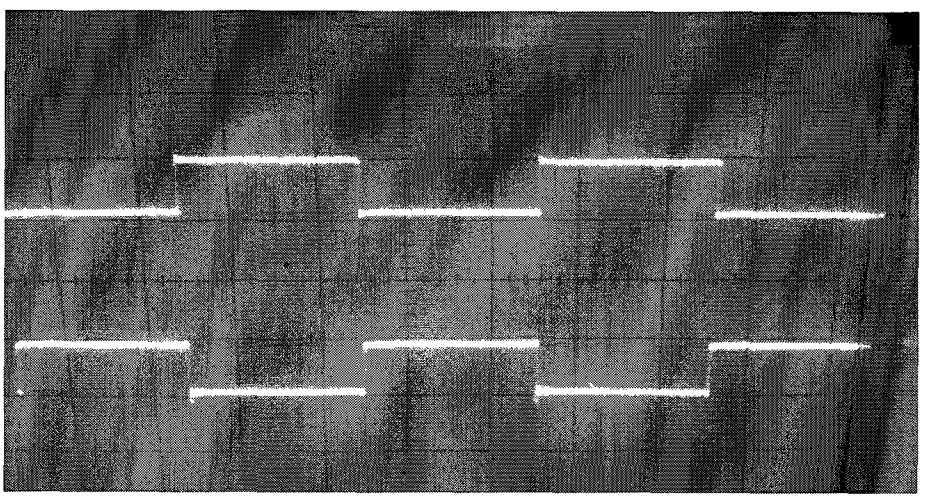

Figure 5-9 Clock Signal (Bottom) Inverter Output (Top)

The retriggerable monostable multivibrator is used to generate a programmable pulse-width of 1 μs. It generates a pulse at the falling edge of the input clock to trigger the loading of the exponent in the D Flip-Flop. The oscilloscope output for the monostable is shown bellow.

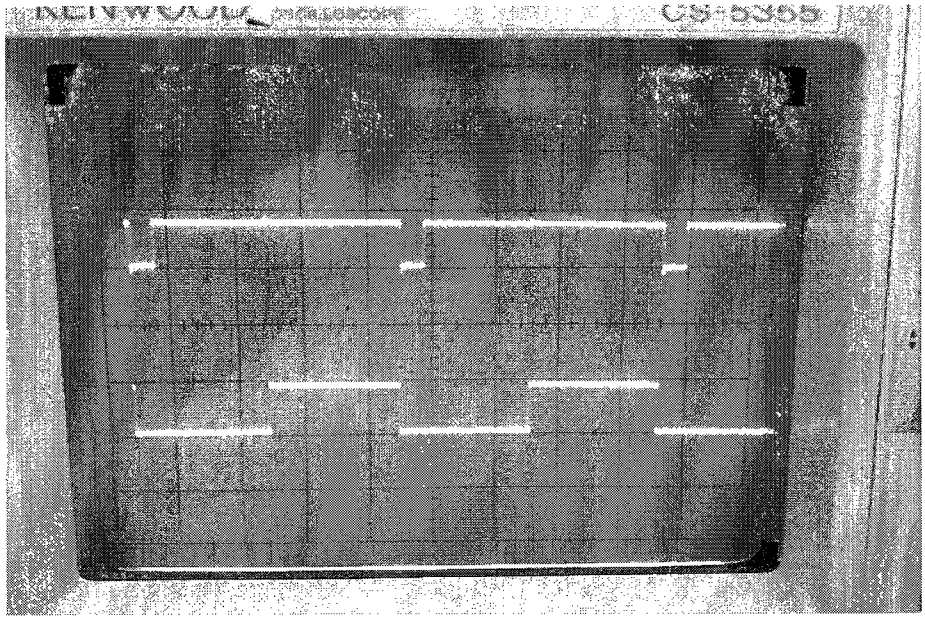

Figure 5-10 Clock Signal (Bottom) Monostable Output (Top)

The AD585 is a complete monolithic sample-and-hold circuit. Its performance makes it ideal for data acquisition system. The next photograph of the oscilloscope shows the SH control command at the top and the SH output.

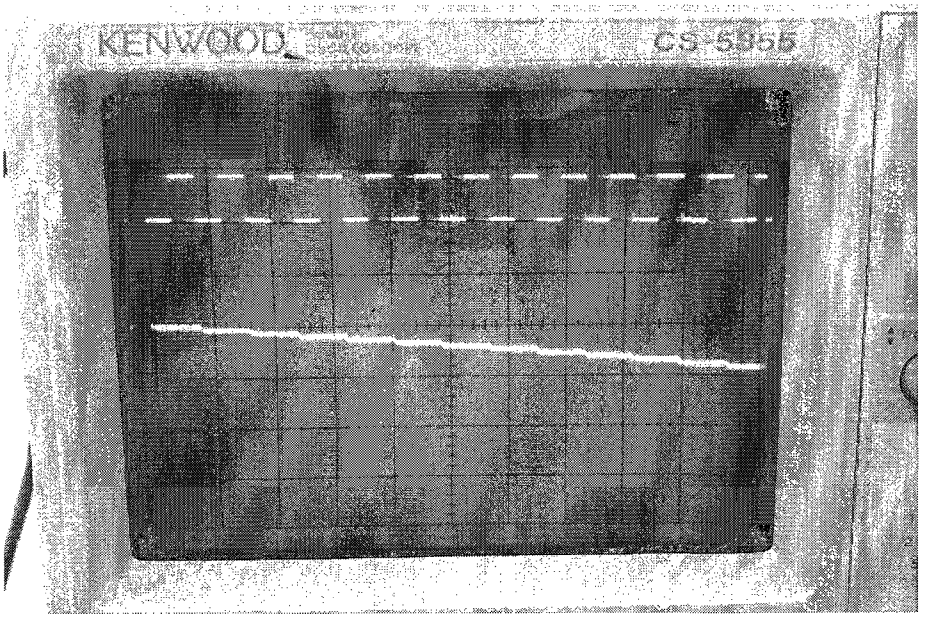

Figure 5-11 Clock Signal (Top) SH Output (Bottom)

The first SH output as it tracks the input signal Vin is depicted in Figure 5-12.

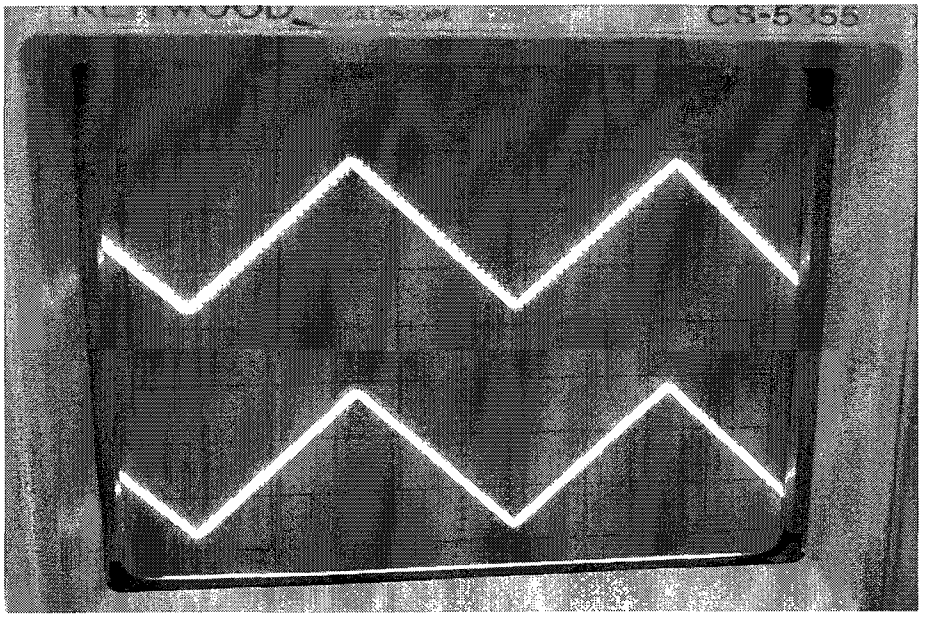

Figure 5-12 SH1 Output (Top) Input Signal (Bottom)

Similarly the second SH output as it tracks the output signal of the programmable gain amplifier is depicted in Figure 5-13.

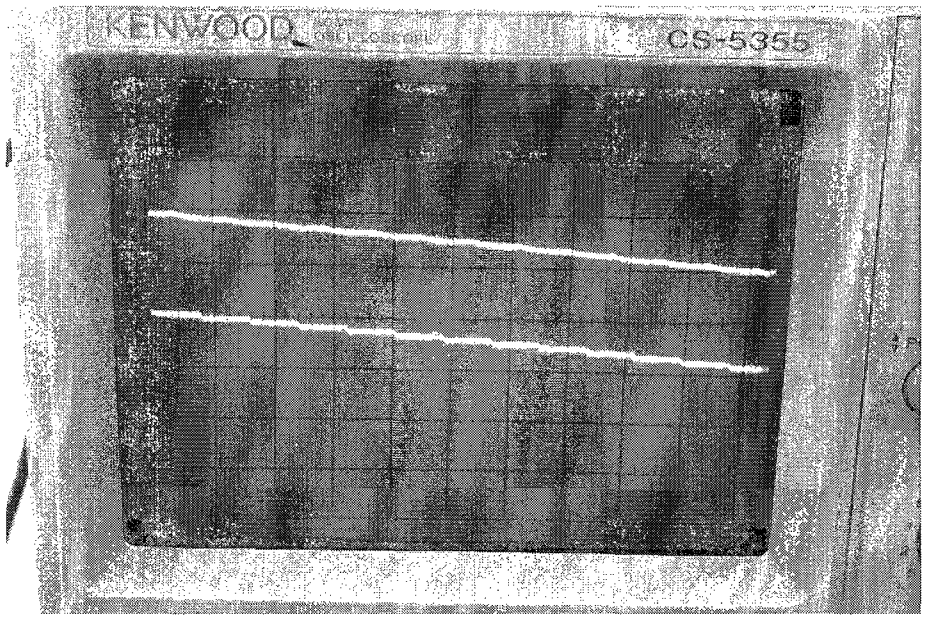

Figure 5-13 SH2 Output (Bottom) PGA Output (Top)

The AD526 is a software gain amplifier. A particular gain is selected by applying the appropriate gain to the control logic according to Table 5-1.

Table 5-1 Gain Codes

V_{out}/V_{in}	A2	A1	A0	B
1	0	0	0	1
2	0	0	1	1
4	0	1	0	1
8	0	1	1	1
16	1	0	0	1

When $\overline{CS}$ and $\overline{CLK}$ are held high by the monostable, the AD526 gain state will remain constant regardless of the transitions at the A2, A1, A0 and B inputs. The next

picture shows the output of the PGA. It can be seen that the signal is amplified while the gain is adjusted so that it stays in the upper-scale of the input range.

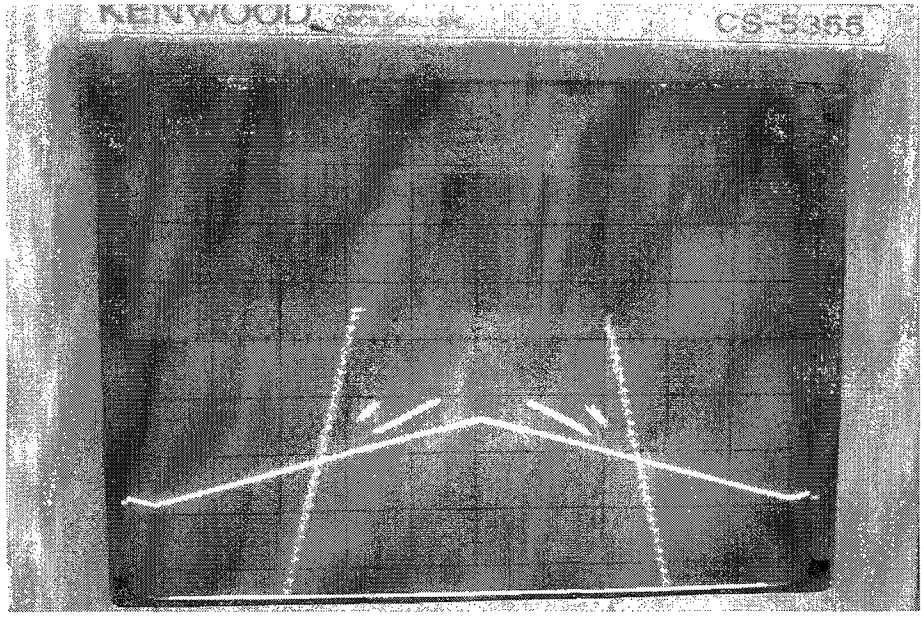

Figure 5-14 PGA Vin (Bottom) and Vout (Top)

The AD7572 12-bit ADC (mantissa) status is indicated by the $\overline{BUSY}$ output, and this is low while the conversion is in progress. It is shown in Figure 5-15 at the top while the control signal which consists of three digital inputs (HBEN, $\overline{CS}$ and $\overline{RD}$) is shown at the bottom. The conversion time of the AD7572 is roughly 5μs. Note that the INL for the AD7572 is ±1/2 LSB, the DNL is ±1 LSB and the offset error is ±3 LSB.

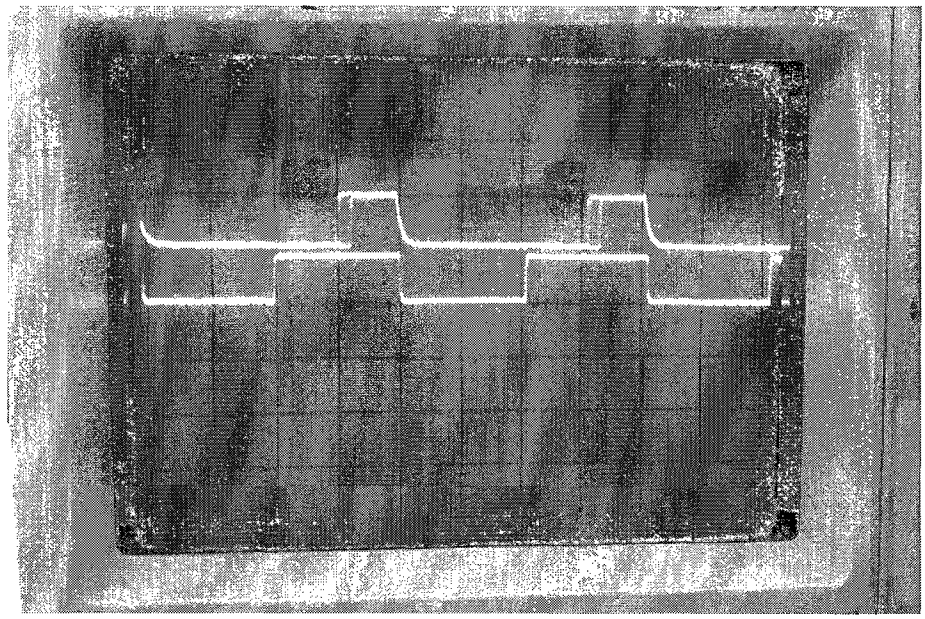

Figure 5-15 Conversion status of the A/D-M, Clock Signal (Bottom), End of Conversion Signal (Top)

The flash auto-ranger consists of four comparators (the LM339) connected in parallel with reference voltages set by a resistor network. Each comparator produces "1" when its analog input voltage is higher than the reference voltage applied to it. These reference voltages are theoretically 2.5V, 1.25V, 0.625V and 0.3125. In real world these values are approximately 2.62V, 1.242V, 0.873V and 0.576V as shown in Figure 5-16.

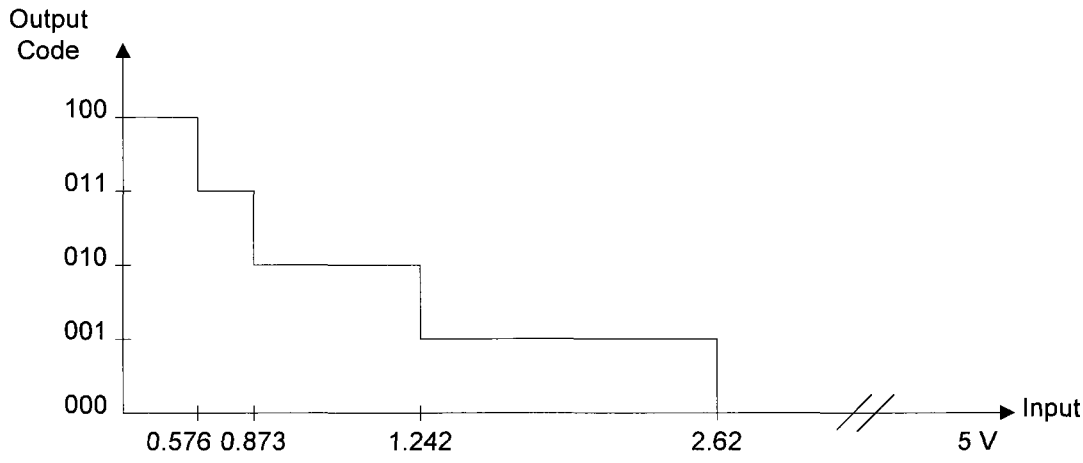

Figure 5-16 Ideal Input/Output Transfer Characteristic

The embedded logic analyzer within the Altera Quartus software is a system-level debugging tool that captures and displays real-time signal behavior. It reads the content of the memory where the Mantissa and Exponent are stored. Figure 5-17 shows the logic analyzer output which is the exact same output of the FP-ADC.

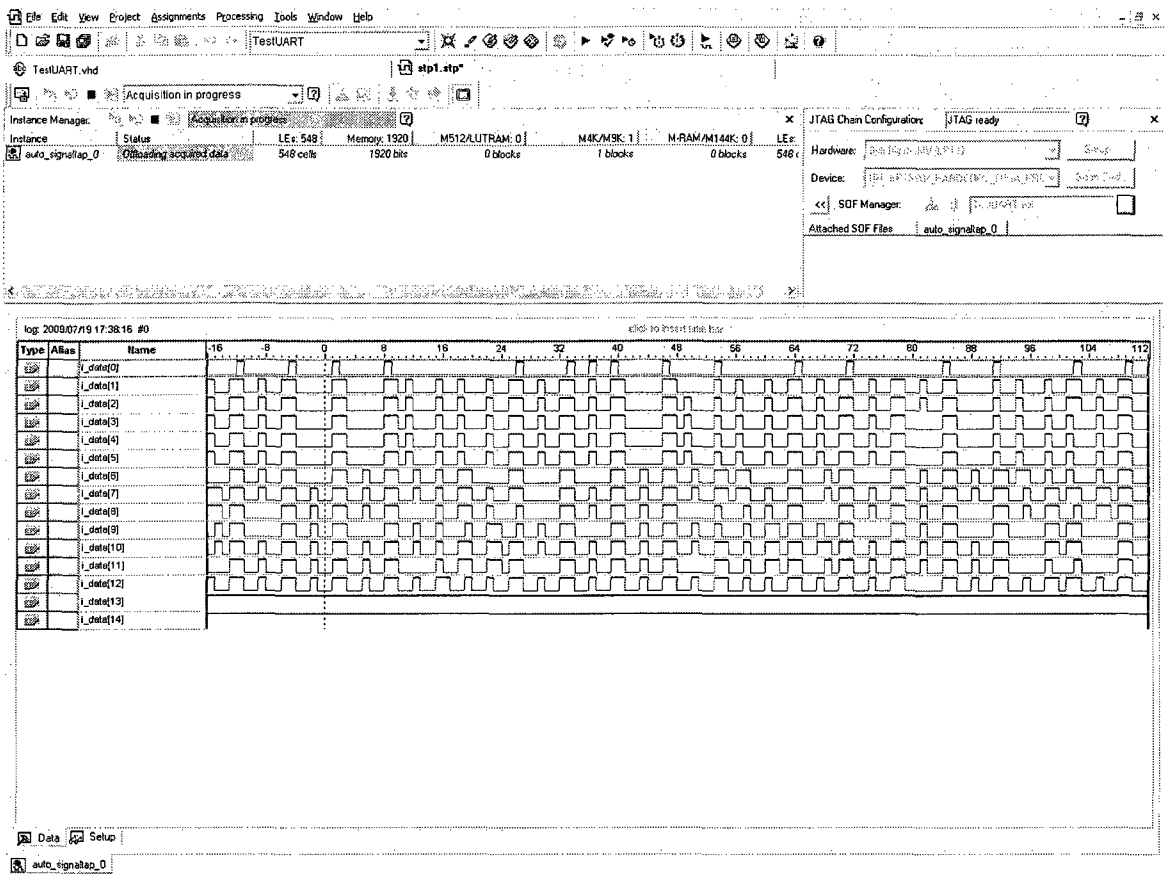

Figure 5-17 Logic Analyzer Output

5.2. FPGA Subsystem Simulation and Testing

The methodology used for synthesizing the Register Transfer Level FPGA design is the bottom-up approach. The top-level module TestUART generated at the final stage of this approach is subdivided into 12 sub-modules each was tested thoroughly by mean of functional simulations before being incorporated into larger blocks formed by the UART, Memory and Control Unit. This ensures the correctness of each component facilitates the integration process and eases the design flow. The top-level design is finally validated by testing its real operation.

The first module under test is the UART because it is responsible for the serial communication between the FPGA and the serial communication port of the PC. It is necessary for this component to work as designed so that the control commands can be sent to the FPGA and the exponent and mantissa acquired from the FP-ADC transmitted to the PC. The UART is break down in the receiver, transmitter and baud rate generator modules presented bellow.

5.2.1. Transmitter Simulation

In the following simulation 01000110 at the transmitter input is being sent at a Baud clock speed. When the signal RTDV = '0' (transmitter register full) is received the byte is sent serially in the TxD pin and the RTDV output is set back accordingly to '1' (transmitter register is empty).

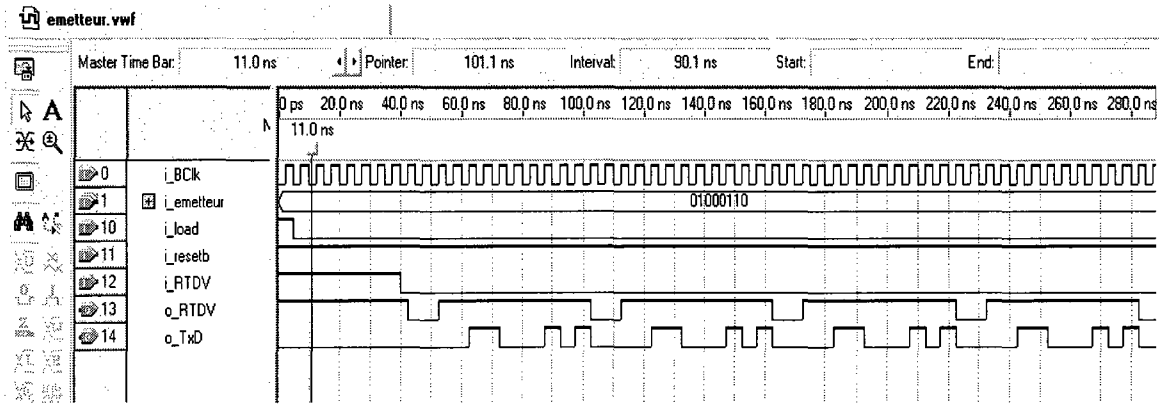

Figure 5-18 Transmitter Simulation

The next simulation shows the operation of the FSM transmitter. It can be seen that when RTDV turns to logic "0", that is the data transmission register is full, the state machine moves from state A to state B in which the counter is started and the shift

register is loaded with the data in parallel (o_load = '1'). The next state C is for shifting the data and transmitting them serially. Note that the control signal RTDV is set to '1'. After 9 clock cycle the data and the stop bit are transmitted and the state machine is back on state A.

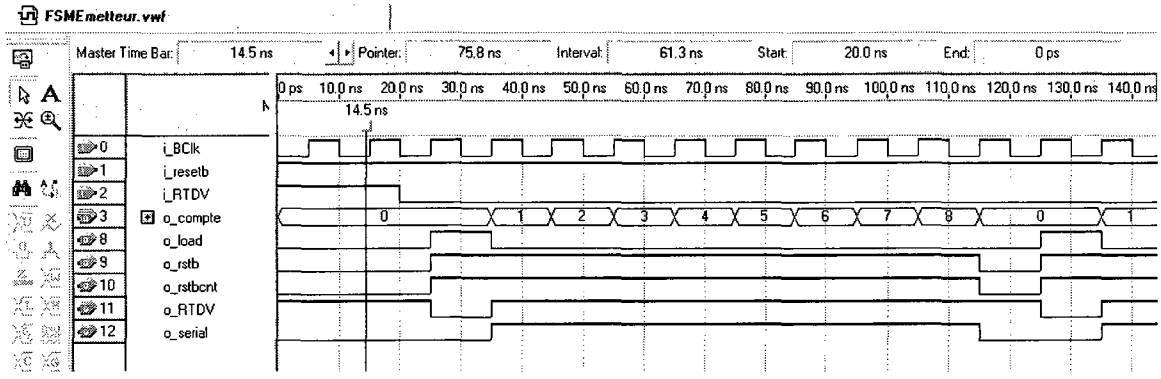

Figure 5-19 FSM Transmitter

5.2.2. Receiver Simulation

The simulation of the receiver bellow shows the byte '00110011' being received successfully at the pin RxD as it appears at the output of the data reception register (o_recepteur).

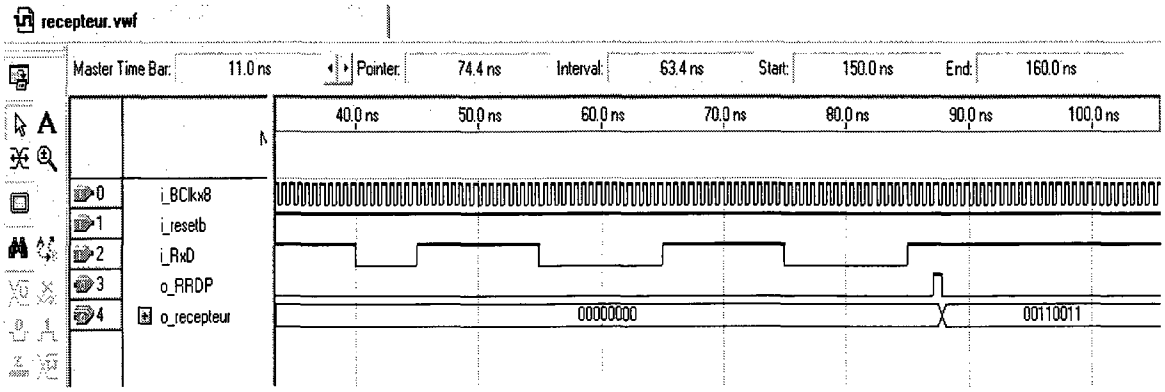

Figure 5-20 Receiver Simulation

The simulation of the FSM receiver that controls the receiver behavior is shown next. When the start bit is received (RxD is "0") the first counter is started. After 3 BClkx8 periods, this means the middle of the start bit, the second counter is started and every 8 BClkx8 periods the bits are serially shifted (Decale = "1") in the eight bit register RDRD. After another 8 BClkx8 periods, the state machine transition to the next state E where the stop bit is received and not shifted in RDRD. In the next 8 BClkx8 periods, the register RRD is loaded with the content of RDRD (0_loadRRD = '1') and the status bit RRDP for the data reception register full is set to "1". At this point, the state machine returns to the default state A in the next BClkx8 transition.

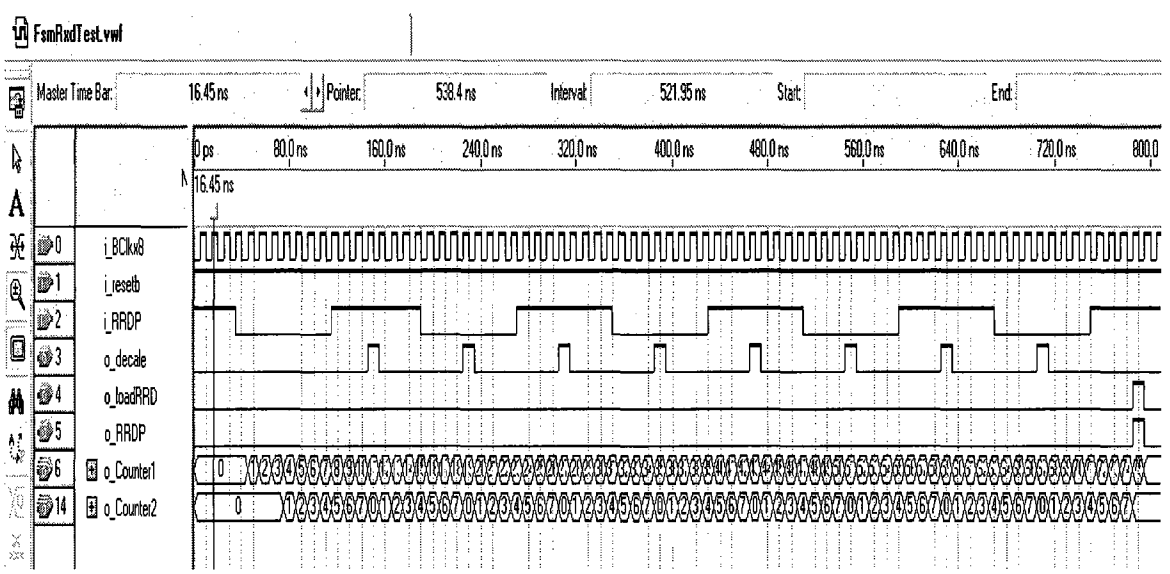

Figure 5-21 FSM Receiver

The simulation of the baud generator shows the baud clocks being set according to the selector Sel. BClk is the input clock divided by 344 and BClkx8 is the input clock divided by 43 or 8 times faster than BClk.

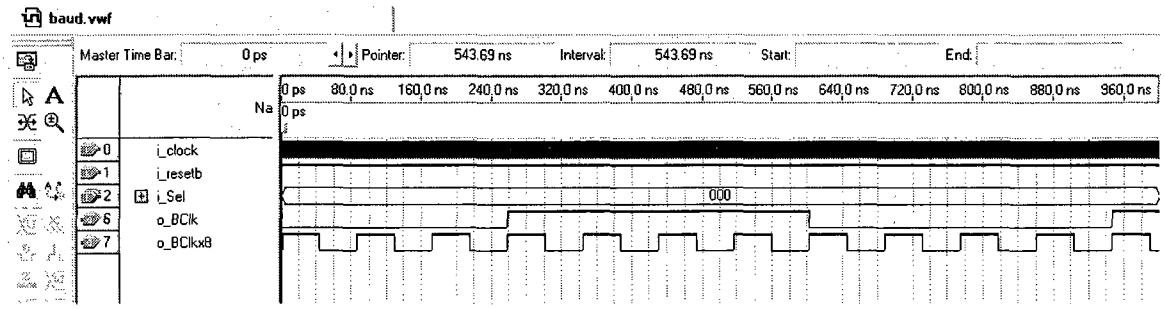

Figure 5-22 Baud Generator Simulation

In the last simulation of the UART, the receiver is connected to the transmitter in a loop. The highlighted 10 bits (1 start bit, 8 bits of data and 1 stop bit) in are identical at the Rxd and the TxD pins which proves the proper operation of the UART.

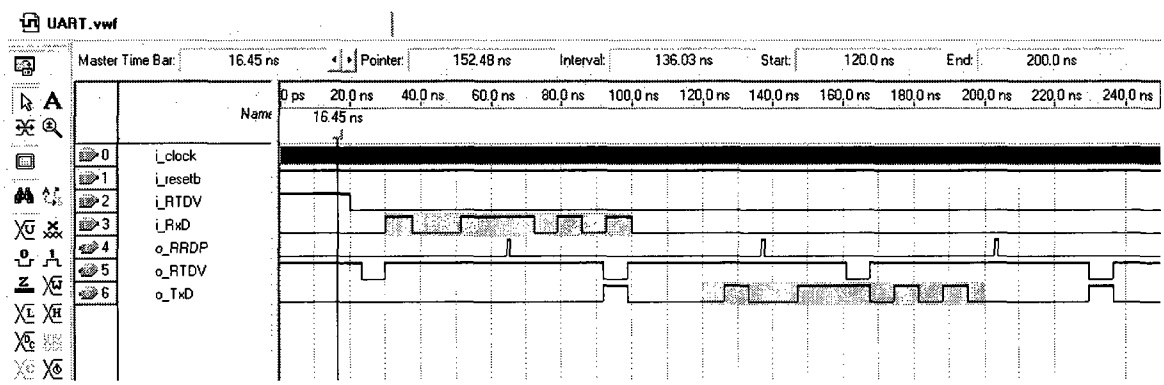

Figure 5-23 UART Simulation

5.2.3. Memory Simulation

The next component under test is the memory block. The RAM is initialized in the Memory Initialization File (MIF) which specifies the initial content of the memory. In this simulation, the memory is 8 bits x 32 words. The RAM is instantiated with

hexadecimal numbers ranging from 00 to 1F. When a write command is enabled (wren ='1'), number 14 is written at the location pointed out by the write address '7'. When the read command is enabled, the content of the memory is read.

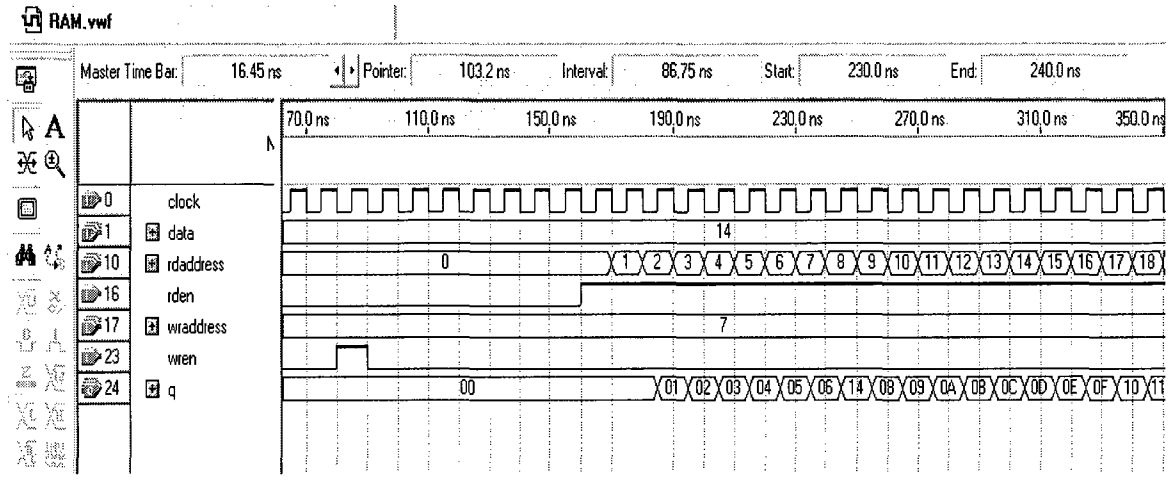

Figure 5-24 RAM Simulation

5.2.4. Control Unit Simulation

The control Unit controls the data flow between the daughter card, the FPGA and the PC. When the start signal is received, the state machine moves to the writing state B. In this state wren is enabled which means the writing process on the RAM is started. The counter is also started and the writing continues until it runs out of memory space when it reaches the last index indicated by the counter which is in this simulation 15. Then the content of the memory is read and ready to be transmitted in state D. If the data transmission register is full the reading process stops. Finally, when all of the content of the memory is read the state machine return to the idle state A. the simulation of the control Unit is shown in Figure 5-25.

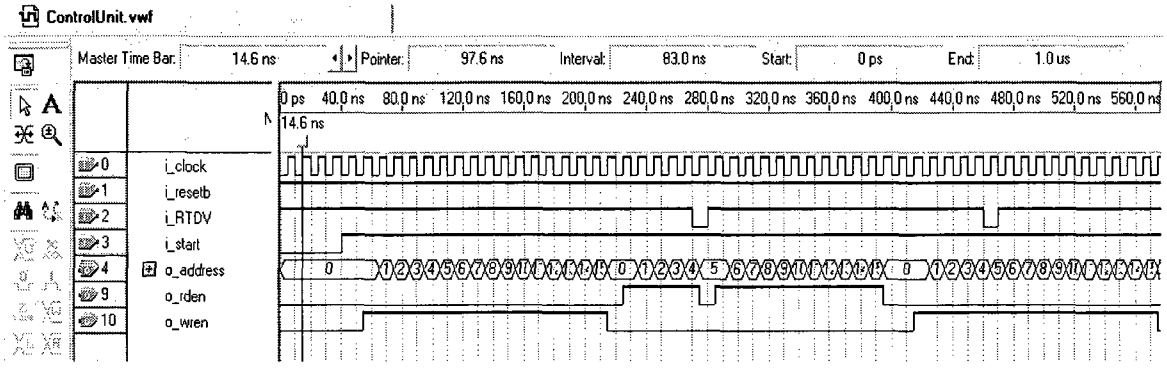

Figure 5-25 Control Unit Simulation

At last, the operation of the three components (UART, Memory and Control Unit) is tested. The Character "W" is written in the first four addresses of the memory. Then the content of the memory which is previously initialized is read and transmitted to the PC terminal through the UART at the reception of the Start command ("S"). The result is shown in the Figure 5-26 . The serial port is set up on COM1 at 57600 baud, 8 data bits, 1 start bit, 1 stop bit, no parity bit and no handshake.

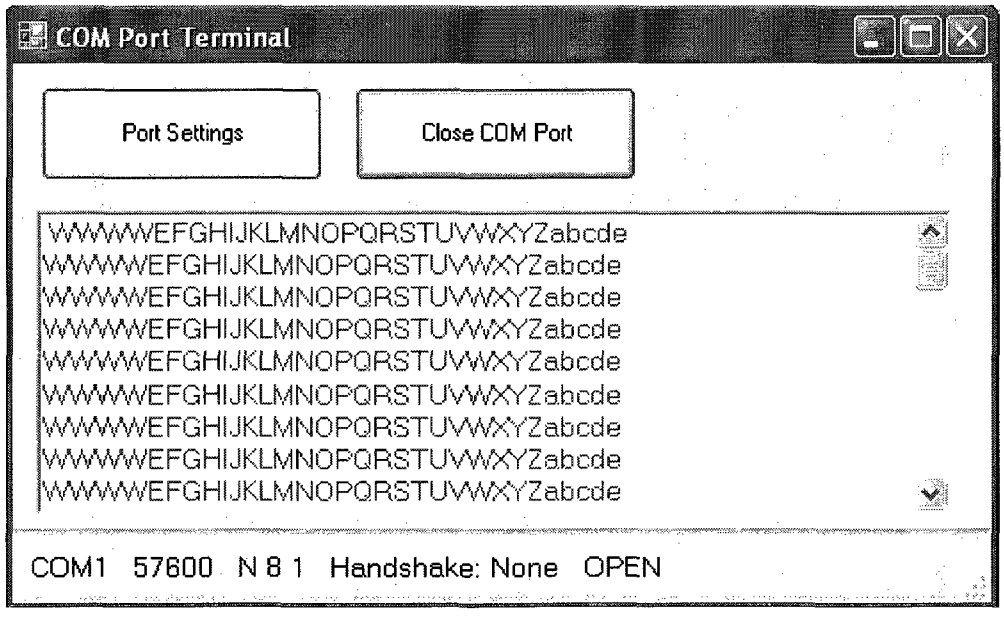

Figure 5-26 FPGA Subsystem Testing

5.3. Results and Conclusion

A new architecture of FP-ADC is presented, designed, simulated, implemented and tested in the thesis along with a FP-ADC prototyping system on a PCB using off the shelf components and real world hardware.

The A/D converter has a dynamic range of 96 dB. The dynamic range of a converter is the ratio of the full-scale input range to the LSB value. With a floating-point A/D converter, the smallest value LSB corresponds to the LSB of the monolithic converter divided by the maximum gain of the PGA. The floating-point A/D converter has a full scale range of 5V, a maximum gain of 16V/V from the AD526 and a 12-bit A/D converter; this produces:

$$LSB = \left(\left[FSR / 2^{N}\right] / Gain\right) = \left(\left[5V / 4096\right] / 16\right) = 76\mu V \qquad (5\text{-}1)$$

The dynamic range (DR) in dBs is based on the log of the ratio of the full-scale input range to the LSB.

$$DR = 20\log\left(5V / 76\mu V\right) = 96dB \qquad (5\text{-}2)$$

The obtained test results indicate that the system exhibits the improved characteristics of the FP-ADC. The resulting quantizing resolution is higher for smaller input signals, which in turn maximizes the SNR of the system. The results also point out the weakness of the FP-ADC and suggest an array of future improvements.

Some problems of system noise and linearity inherent to mixed-signal circuit could arise in the current concept. However, good design practices such as physically separating the analog components from the digital ones and using bypass capacitors when required to decouple the integrated circuits from the power supply line were implemented during the prototyping phase. As a consequence, the FP-ADC did not appear to be

particularly vulnerable to undesirable interferences during the testing phase. In the case of higher frequencies though, further shielding work should be undergone with copper pours around the components to help protect against switching noise and voltage peaks.

Chapter 6

6. Conclusions and Future Work

This chapter is the conclusion of the thesis and the proposed future work.

6.1. Conclusions

This research work conceived and evaluated a new sequential architecture of the floating-point analog-to-digital converter implemented on a printed circuit board (PCB). This 2-step architecture achieves a high throughput rate of 125 KHz by combining two sample-and-hold (SH) amplifiers and a fast A/D converter. The sampling interface is optimized by overlapping the acquisition time of the first sample-and-hold amplifier and the settling time of the gain amplifier with the conversion time of the A/D converter. The second SH amplifier holds the amplified signal while the A/D converter perform its conversion routine. The proposed solution takes advantage of two A/D converters that work in tandem; one determines the exponent (A/D-E) which is represented within 4-bit, while the second one which is connected to the quantizer input over a programmable gain amplifier (PGA) finds the mantissa (A/D-M) which is represented within 12-bit for a total of 16-bit outputted data for the FP-ADC.

Part of the work also included the FPGA (Altera Stratix) system which implements the serial communication interface between the sequential architecture of the FP-ADC and a PC.

6.2. Contribution of the thesis

This thesis made the following contributions

- Proposing of a novel sequential FP-ADC concept
- Design of the FP-ADC
- Implementation of the FP-ADC.
- Devising a methodology for the FP-ADC

This dissertation has shown that the sequential floating-point ADC architecture exhibited and provided an increase in resolution and overall dynamic range. A number of improvements can still be made some of which are covered in the next section.

6.3. Future work

Several areas can be explored in the future in order to expand and improve the research on the FP-ADC described in this thesis.

The sequential FP-ADC as implemented does not need any computing power and is a completely stand-alone solution, the parallel implementation of the FP-ADC on the other hand uses a Gain Control Unit on the FPGA side to predict the exponent in the next sample as described in section 3.2 and would eventually halve the conversion time. This architecture relies mostly on the goodness of the predictive algorithm responsible on setting the gain for the PGA when calculating the mantissa. It is believed that a more sophisticated algorithm for the gain prediction in comparison to a simple extrapolation used so far will improve on the number of hits.

The signal noise is a major concern in the FP-ADC. The noise is inherent to mixed-component circuitry in the FP-ADC design on the daughter card. The next improvement will come from a better noise insulation of the components combined with

an implementation of some noise attenuating solutions like filters. This will ensure that the data acquisition system preserves the integrity of the signal.

References

[1] Shumin Shen, "A Floating-Point Analog-to-Digital Converter", Master Thesis Performed at the School of Graduate Studies and Research of the University of Ottawa, 2004

[2] J. Yuan, J Piper, "Floating-Point Analog-To-Digital Converter", Electronics, Circuits and Systems, 1999. Proceedings of ICECS '99. The 6th IEEE International Conference on. 02/1999; 3:1385-1388 vol.3.

[3] J. Yuan, J Piper, "Realization of a Floating-Point A/D Converter", ISCAS 2001 IEEE International Symposium on Circuits and Systems, vol. 1, pp. 404–407, Sydney, Australia, 2001.

[4] V. Z. Groza, "Floating-Point Data Acquisition System with Improved Noise Immunity", IEEE Instrumentation and Measurement Technology Conference IMTC 2003, ISBN 0-7803-7705-2, pp. 1454-1458, Vail, Colorado, 20-22 May, 2003

[5] V. Z. Groza, "High Resolution Floating-Point Analog-to-Digital Converter", IMTC1999, Proceedings of the 16th IEEE, vol.3, Instrumentation and Measurement Technology Conference, pp 1663-1666.

[6] Gene Frantz, Ray Simar "Comparing Fixed- and Floating-Point DSPs", Texas Instruments

[7] Steven W. Smith "The Scientist and Engineer's Guide to Digital Signal Processing", 1997. Visit the book's website at: www.DSPguide.com

[8] http://en.wikipedia.org/wiki/Digital_signal_processor

[9] The Institute of Electrical and Electronics Engineers, "IEEE standard for terminology and test methods for analog-to-digital converters" IEEE Std 1241-2000, New York, NY, USA, 2000.

[10] Walt Kester, "The Data Conversion Handbook", Elsevier/Newnes, 2005, ISBN 0-7506-7841-0, Chapter 2.

[11] David F. Hoeschele, "Analog-to-Digital and Digital-to-Analog Conversion Techniques" Wiley-Interscience, 1994, ISBN 0-471-57147-4, Chapter 1.

[12] Sergio Rapuno, Pasquale Daponte, Eulalia Balestrieri, Luca De Vito, Steven J. Tilden, Solomon Max, Jerome Blair, "ADC Parameters and Characteristics", IEEE Instrumentation and Measurement Magazine, volume 8, no.5, pp 44-54, December 2005.

[13] The institute of Electrical and Electronics Engineers, "IEEE standard for binary floating-point arithmetic" IEEE Std 754-1985, New York, NY, USA, 1985.

[14] F. Maloberti, "High-Speed Data Converters for Communication Systems", IEEE Circuits and Systems, ISSN 1531-663X, vol.1, no.1, pp.26-36, 2001.

[15] Widrow, B., Kollar, I. and Liu, M.-C.,"Statistical theory of quantization," IEEE Transactions on Instrumentation and Measurement, volume 45, no.2, pp 353-361, Waltham, MA, USA, April 1996.

[16] Syed Arsalan Jawed, "Analog-to-Digital Converter Design for Non-Uniform Quantization" Master Thesis, Fraunhofer Institute, Germany and Electronic Devices Department, Linkoping University, Sweden, 2004

[17] http://www.imec.be/esscirc/ESSCIRC2002/presentations/Slides/C32.02.pdf

[18] http://www.mit.bme.hu/books/quantization/floating-point.pdf

[19] V. Z. Groza, "High Resolution Floating-Point ADC", IEEE Transactions on Instrumentation and Measurement, Vol.50, No.6, pp1822-1829, Baltimore, MD, USA, December 2001

[20] V. Z. Groza, "Floating-Point ADC Optimized for Acquisition of Deterministic Signals", IEEE Instrumentation and Measurement Technology Conference, IMTC 2002, ISBN 0-7803-7218-2, pp. 707 – 712, Anchorage, Alaska, 21-23 May, 2002

[21] V. Z. Groza, "Floating-Point Analog-to-Digital Converters with Predictive Auto-ranging", IEEE Instrumentation and Measurement Technology Conference, IMTC 2000, vol. 2, pp. 759–762, Baltimore, MD, USA, May 2000,

[22] Bingxin Li; Tenhunen, H, "A second order sigma delta modulator using semi–uniform quantizer with 81dB dynamic range at 32x OSR", Proceedings of the 28th European Solid-State Circuits Conference, 2002. ESSCIRC 2002, Volume, Issue, Page(s): 579 – 582, Firenze, Italy, 24-26 Sept. 2002,

[23] S. Sharma, G. Otomo, K. Tsukamoto, and T. Miyata, "A floating-point A/D converter uses low resolution DAC to get wide dynamic range", Int. J. of Electronics, vol. 64, pp. 787–794, May 1988. 36, 38

[24] G. Ootomo, K. Tsukamoto, T. Watahiki, and T. Miyata, "A floating point A/D converter with self-calibration" Trans. of the inst. of Electronics, Information and Communication Eng., vol. E71, pp. 1303–1308, 1988. 36, 38

[25] J. Piper "Floating-Point Analog-to-Digital Converter", Ph.D Thesis performed at the Competence Center for Circuit Design at the Department of Electroscience, Lund University, Sweden, 2004.

[26] V. Groza, B. Dzerdz, "FPGA Based Implementation of a Floating-Point Analog-to-Digital Converter", 4-th IEE International Conference on Advanced A/D and D/A Conversion Techniques and their Applications & 7-th European Workshop on ADC Modelling and Testing ADDA&EWADC 2002, ISBN 80-01-02540-3, pp.143-146, Prague, Czech Republic, June, 2002

[27] W. Kester, "Which ADC architecture is right for your application?" , Analog Dialogue, vol.39, no.6, June 2005 available at: http://www.analog.com/library/analogDialogue/cd/vol39n2.pdf#page=11

[28] J. Kontro, K. Kalliojarvi, and Y. Neuvo, "Floating-point arithmetic in signal processing," in IEEE International Symposium on Circuits and Systems, vol. 4, pp. 1784–1791, San Diego, CA, USA , May 1992.

[29] D.U. Thompson, B.A. Wooley, "A 15–bit pipelined floating–point A/D converter", Proceedings of the 25th European Solid-State Circuits Conference Volume, Issue, 21-23 Sept. 1999 Page(s): 170 – 173.

[30] Altera Corporation, "Stratix EP1S80 DSP Development Board Data Sheet", available at: http://www.altera.com/literature/ds/ds_stratix_dsp_bd_pro.pdf

[31] Analog Devices, "AD526 Software Programmable Gain Amplifier Data sheet", Page12, available at: http://www.analog.com/static/imported-files/Data_Sheets/AD526.pdf